



Universidad
Carlos III de Madrid

Departamento de Informática
Ingeniería en Informática

PROYECTO FIN DE CARRERA

Infraestructura móvil para el despliegue de servicios basados en la localización en la Universidad Carlos III de Madrid (Plataforma iOS)

Autor: D. Adrián Casado Rivas

Tutor: Dr. Javier García Guzmán

Leganés, 29 de octubre de 2012

Título: Infraestructura móvil para el despliegue de servicios basados
en la localización en la Universidad Carlos III de Madrid
(Plataforma iOS)

Autor: D. Adrián Casado Rivas

Director: Dr. Javier García Guzmán

EL TRIBUNAL

Presidente: Dr. Antonio de Amescua Seco

Vocal: D. Jesús Gago Mejías

Secretario: Dr. Alberto Heredia García

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día 29 de octubre de 2012 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE

Agradecimientos

Es en los momentos difíciles y en los de superación personal cuando una persona se da cuenta quiénes están realmente con él, pero sería egoísta no querer compartir con aquellos que le apoyan los buenos momentos de su vida. A lo largo de mi vida personal y académica he tenido la oportunidad de conocer a muchas personas y de todas, me quedo con algunas, aunque todas ellas me han hecho crecer como persona porque de todas ellas algo he aprendido que sin duda pasará a formar parte de mi experiencia de vida.

Todo el mundo lo hace, pienso que algunas veces la mayoría puede equivocarse, pero ésta es una de las veces que tiene razón: hay que comenzar agradeciendo el haber llegado a cerrar una etapa importante en la vida de un estudiante a la familia. A la mía le estoy muy agradecido por su apoyo incondicional y por los buenos consejos que tanto mi madre, mi padre y mi hermano me han dado durante todo este tiempo. Estoy seguro de que los voy a seguir recibiendo y anticipadamente me disculpo con vosotros porque sabéis que voy a continuar estudiando y bueno, hay momentos que en los que soy difícil de llevar.

No me arrepiento de nada de lo que he hecho durante estos cinco años en la universidad, al final se han quedado cortos. Estoy muy orgulloso de las amistades que he ganado y por partida doble. Por un lado, os encontráis mis amigos de clase, los que juntos hemos conseguido llegar hasta el final. Me alegra mucho haber compartido la carrera con todos los compañeros porque eso es lo que ha primado, el compañerismo. Ya lo sabéis todos, nuestra titulación se extingue: somos la resistencia. De entro todos me gustaría mencionar a Álvaro, Rober, Coral, Laura, Dani, Fran, Sofía y Luis. Sois unos grandes de este oficio.

Por otro lado, como consecuencia de mis inquietudes, no sólo he pasado estos cinco años estudiando, sino que los he pasado en la que ha sido mi segunda casa y en la que mucho de los que estamos allí tenemos una cama, aunque sea imaginaria: he tenido la gran suerte de pertenecer a la Delegación de Estudiantes de la EPS. Allí he encontrado a personas con mis mismas inquietudes y ha sido un lugar de cobijo ya que las puertas de todos los compañeros de allí siempre están abiertas. Sin duda, la experiencia que he adquirido allí me ha hecho crecer mucho como persona y me ha dado una perspectiva de cómo puede ser el mundo ahí fuera. He de agradecer a Luis que me terminara de convencer para entrar de lleno y a él y a Rodri todo lo que me han enseñado. También agradecer a los compañeros que estaban cuando entré, a los que han estado conmigo y a los que ahora van a tomar el relevo vuestro trabajo por y para los estudiantes. No podría dejar de mencionar a Irene y a Mario que sin duda representan el espíritu de esta Delegación. Se me hace difícil no mencionarlos a todos, sois muchos los que tanto me habéis aportado que tendría que hacer un proyecto especial. Ya sabéis quiénes sois.

En el último año de mi carrera he comenzado mi vida laboral en el grupo de investigación SEL de la UC3M. No os quiero hacer la pelota, pero desde que tomé la primera asignatura de vuestro grupo en la carrera me fijé en él como uno de los sitios donde quería trabajar. En este año y medio con vosotros he aprendido que el valor de las personas es el más importante en el trabajo. Gracias a Javier, mi tutor, por hacerme partícipe de miSEL y por el buen trato que recibo de él al igual que de Antonio, nuestro gran jefe. Especial mención a Juan Carlos, has sido una de las claves que me ha permitido sacar este proyecto adelante. Y por supuesto, Javi y Alberto, no me olvido de vosotros.

En definitiva, todos habéis sido partícipes de mi éxito y por ello os estoy muy agradecido. Gracias.

*“Dicen que los ojos son el reflejo del alma,
pero la música que escuchas es el reflejo de tu vida.”*

Resumen

El problema que pretende resolver el presente proyecto fin de carrera es la localización en interiores, también denominada localización *indoor*, de lugares y personas en la Universidad Carlos III de Madrid.

Se ha detectado que los miembros de la comunidad universitaria de la Universidad Carlos III de Madrid tienen dificultades a la hora de encontrar la ubicación de un lugar, como aulas, laboratorios o despachos, y recordar dónde desempeña su trabajo el personal perteneciente a los colectivos PDI y PAS. Estas dificultades se presentan en diferentes etapas de una persona en su relación con la Universidad ya que el perfil de los potenciales usuarios varía entre personas recién llegadas a la institución y personas que ya la conocen y que necesitan en un momento puntual saber la ubicación de un lugar o de una persona.

La motivación principal para la realización del presente proyecto es proveer de un servicio de localización *indoor* a las personas que en algún momento de su relación con la Universidad Carlos III de Madrid, necesiten conocer la ubicación de un lugar o de una persona y no sepan cómo encontrarlo. Para que la aplicación resultante del proyecto pueda ser utilizada en cualquier contexto, se ha optado por su desarrollo para las plataformas móviles, siendo iOS el entorno de desarrollo elegido.

El desarrollo del proyecto se enmarca dentro de la línea de investigación de mejora de la vida universitaria abierta por el grupo de investigación Software Engineering Lab, SEL, de la Universidad Carlos III de Madrid. Los objetivos establecidos para la consecución del proyecto han sido el desarrollo de un servicio web de localización para la UC3M, la georreferenciación de los campus de la Universidad Carlos III de Madrid y la adaptación de los planos de sus edificios y finalmente, el desarrollo de una aplicación iOS para proveer servicios de localización *indoor* a su comunidad universitaria.

Los resultados de la realización del proyecto han sido satisfactorios ya que se ha conseguido completar la totalidad de los objetivos con la participación de equipos multidisciplinares de la Universidad Carlos III de Madrid.

Una vez distribuida la aplicación iOS, se ha podido medir el nivel de aceptación y el uso que los usuarios hacen de la misma gracias a un sistema *dashboard* que permite monitorizar el número de consultas realizadas por los usuarios facilitando los datos agregados del total de búsquedas y los datos desagregados del tipo búsqueda además de ofrecer información del número de consultas únicas por usuario.

La realización del proyecto ha sido un reto que ha resultado en la inmersión en el conocimiento de las dificultades del posicionamiento *indoor*, en el aprendizaje en el desarrollo de aplicaciones para dispositivos móviles y en la experiencia de trabajar directamente en un entorno de investigación real en cooperación con otros equipos de disciplinas logrando añadir un valor añadido al proyecto fin de carrera.

Abstract

The issue that this final degree project wants to solve is the indoor location of places and people at Carlos III of Madrid University.

It has been detected that members of Carlos III of Madrid University community college have difficulties at the moment of find the location of a place, such as classrooms, labs or offices, and remember where a person that belongs to the PDI or PAS group performs his daily activity. These difficulties appears on different stages on the relationship of a person with the University because the profile of the potential users range between newcomers of the institution and people that is maintaining a long relationship with it and need to know in a specific moment the location of a place or of a person.

The main motivation to carry out this project is to provide an indoor location service to the people that, in some moment of his relationship with Carlos III of Madrid University, need to know the location of a place or a person and do not know how to find it. With the purpose that the application product of the project can be use in some context, it has been decided to develop it for mobile platforms. The chosen development environment has been iOS.

The development of the project is enshrined on the improvement of the university life research line opened by the Software Engineering Lab, SEL, of Carlos III of Madrid University. The aims established for the project development had been the development of a location web service for the UC3M, the georeferentation of Carlos III of Madrid University campuses and the adaptation of its planes and finally, the development of an iOS application to provide location services to its community college.

The results of the development of this has been satisfactory because of it has been achieved to complete the whole aims with the participation of multidisciplinary teams of the Carlos III of Madrid University.

Once the application has been distributed, it has been possible to measure the acceptance level and the kind of use that users do with it thanks to a dashboard that allows to monitoring the number of consults done by users. The dashboard provides aggregated data by the total numbers of requests and disaggregated data classified by the sort of search. Additionally, it provides information about of the unique requested made by a user.

The development of the project has been a challenge to the author that has culminated in the immersion on the knowledge of indoor location, in the learn of the mobile applications development and in the gained experience of working on a real research environment with the cooperation with another teams of different disciplines achieving to contribute an added value to the final degree project.

ÍNDICE GENERAL

CAPÍTULO 1	INTRODUCCIÓN	1
1.1	PROBLEMA	2
1.2	MOTIVACIÓN DEL PROYECTO	3
1.3	OBJETIVOS	4
1.4	MÉTODO DE RESOLUCIÓN	4
1.5	TERMINOLOGÍA	5
1.6	CONTENIDO DE LA MEMORIA	7
CAPÍTULO 2	ESTADO DEL ARTE	9
2.1	ANÁLISIS DE APLICACIONES WEB Y MÓVILES RELACIONADAS CON LOS OBJETIVOS ESTABLECIDOS	10
2.2	ENTORNO DE DESARROLLO MÓVIL SELECCIONADO	20
2.3	DESARROLLO ÁGIL COMO MÉTODO PARA REALIZAR EL PROYECTO	28
2.4	CONCLUSIONES DEL ANÁLISIS	32
CAPÍTULO 3	ANÁLISIS DE LA APLICACIÓN	35
3.1	INTRODUCCIÓN	36
3.2	DESCRIPCIÓN GENERAL DEL SISTEMA A DESARROLLAR	36
3.3	HISTORIAS DE USUARIO	43
3.4	REQUISITOS ESPECÍFICOS	47
CAPÍTULO 4	DISEÑO DE LA APLICACIÓN	59
4.1	PAPER PROTOTYPE	60
4.2	ALTERNATIVAS DE DISEÑO	69
4.3	DIAGRAMAS DE COMPONENTES Y DE CLASES	75
4.4	DIAGRAMAS DE SECUENCIA	81
4.5	DISEÑO DE LA BASE DE DATOS	87
4.6	DISEÑO DEL REPOSITORIO DE PLANOS	89
CAPÍTULO 5	PRUEBAS DE LA APLICACIÓN	91
5.1	PRUEBAS UNITARIAS	92
5.2	PRUEBAS DE INTEGRACIÓN	98
5.3	PRUEBAS DE SISTEMA	99
CAPÍTULO 6	CONCLUSIONES Y LÍNEAS FUTURAS	109
6.1	CONCLUSIONES	110
6.2	LÍNEAS FUTURAS	112
BIBLIOGRAFÍA		116
APÉNDICE A	PRESUPUESTO DEL PROYECTO	121
APÉNDICE B	MANUAL DE USUARIO	125

LISTA DE FIGURAS

FIGURA 2.1: GOOGLE MAPS (ANDROID)	12
FIGURA 2.2: APP POINT INSIDE & TRAVEL	13
FIGURA 2.3: APP MYCOEX	13
FIGURA 2.4: APP MALL RAT.....	14
FIGURA 2.5: APP WIFARER.....	14
FIGURA 2.6: APP DE LA UNIVERSIDAD DE GRANADA.....	15
FIGURA 2.7: APP DE LA UNIVERSIDAD DE ALABAMA.....	16
FIGURA 2.8: APP DEL MIT	17
FIGURA 2.9: DIVISIÓN DEL MERCADO DE S.O. MÓVILES	20
FIGURA 2.10: CAPAS DE TECNOLOGÍA QUE COMPONEN EL SISTEMA IOS	24
FIGURA 2.11: CICLO DE VIDA DE UNA APP IOS	26
FIGURA 2.12: DIAGRAMA DE FLUJO DE SCRUM.....	31
FIGURA 4.1: VISTA PRINCIPAL (VERSIÓN 1)	61
FIGURA 4.2: CONFIGURACIÓN (VERSIÓN 1)	61
FIGURA 4.3: RESULTADO DE BÚSQUEDA DE UN LUGAR (VERSIÓN 1).....	62
FIGURA 4.4: AMPLIACIÓN DE RESULTADO DE BÚSQUEDA DE UN LUGAR (VERSIÓN 1).....	62
FIGURA 4.5: VISTA PRINCIPAL CON BÚSQUEDA DE LUGAR SELECCIONADA	63
FIGURA 4.6: INSERCIÓN DE TÉRMINO DE BÚSQUEDA DE UN LUGAR.....	63
FIGURA 4.7: RESULTADO DE BÚSQUEDA DE UN LUGAR.....	64
FIGURA 4.8: DETALLE DE BÚSQUEDA DE UN LUGAR.....	64
FIGURA 4.9: VISTA PRINCIPAL CON BÚSQUEDA DE PERSONA SELECCIONADA	65
FIGURA 4.10: INSERCIÓN DE TÉRMINO DE BÚSQUEDA DE UNA PERSONA.....	65
FIGURA 4.11: LISTA DE RESULTADOS DE PERSONAS ENCONTRADAS.....	66
FIGURA 4.12: INSERCIÓN DE TÉRMINO DE BÚSQUEDA DE UNA PERSONA ESPECÍFICO	66
FIGURA 4.13: RESULTADO DE BÚSQUEDA DE UNA PERSONA.....	67
FIGURA 4.14: DETALLE DE BÚSQUEDA DE UNA PERSONA	67
FIGURA 4.15: VISTA DE CONFIGURACIÓN	68
FIGURA 4.16: VISTA DE ACERCA DE.....	68
FIGURA 4.17: PRIMERA ALTERNATIVA DE DISEÑO.....	72
FIGURA 4.18: SEGUNDA ALTERNATIVA DE DISEÑO	73
FIGURA 4.19: TERCERA ALTERNATIVA DE DISEÑO	74
FIGURA 4.20: DIAGRAMA DE COMPONENTES DE LA APLICACIÓN	76
FIGURA 4.21: DIAGRAMA DE CLASES DE LA APLICACIÓN	77
FIGURA 4.22: CLASES DEL COMPONENTE MAP	78
FIGURA 4.23: CLASES DEL COMPONENTE OPTIONS.....	80
FIGURA 4.24: DIAGRAMA DE SECUENCIA DE INICIO DE LA APLICACIÓN	81
FIGURA 4.25: DIAGRAMA DE SECUENCIA DE BÚSQUEDA DE UN LUGAR	82

FIGURA 4.26: DIAGRAMA DE SECUENCIA DE BÚSQUEDA DE UNA PERSONA.....	83
FIGURA 4.27: DIAGRAMA DE SECUENCIA DE CONFIGURACIÓN	85
FIGURA 4.28: DIAGRAMA DE SECUENCIA DE ACERCA DE	86
FIGURA 4.29: DIAGRAMA ENTIDAD RELACIÓN DE LA BASES DE DATOS	87
FIGURA 4.30: ESQUEMA RELACIONAL DE LA BASE DE DATOS	88
FIGURA 5.1: RESULTADO DE LA BATERÍA DE PRUEBAS DE CONFIGURACIÓN	93
FIGURA 5.2: RESULTADO DE LA BATERÍA DE PRUEBAS DE BÚSQUEDA DE LUGARES	94
FIGURA 5.3: RESULTADO DE LA BATERÍA DE PRUEBAS DE BÚSQUEDA DE PERSONAS	95
FIGURA 5.4: RESULTADO DE LA BATERÍA DE PRUEBAS DE PETICIONES EXTERNAS.....	96
FIGURA 5.5: RESULTADO DE LA BATERÍA DE PRUEBAS DEL DETALLE DE UN LUGAR	96
FIGURA 5.6: RESULTADO DE LA BATERÍA DE PRUEBAS DEL DETALLE DE UNA PERSONA	97
FIGURA 5.7: RESULTADO DE LA BATERÍA DE PRUEBAS DE LA INFORMACIÓN DE LA APP	98
FIGURA 6.1: VISTA PRINCIPAL.....	127
FIGURA 6.2: VISTA DETALLE DE BÚSQUEDA.....	127

LISTA DE TABLAS

TABLA 2.1: COMPARATIVA ENTRE CARACTERÍSTICAS DE LAS APLICACIONES	18
TABLA 2.2: EVOLUCIÓN DEL SISTEMA OPERATIVO IOS.....	23
TABLA 2.3: EVOLUCIÓN DEL SISTEMA OPERATIVO IOS EN APPLE TV	23
TABLA 2.4: METODOLOGÍAS ÁGILES VS. TRADICIONALES	30
TABLA 3.1: CLASIFICACIÓN SEGÚN DISTRIBUCIÓN Y PÚBLICO OBJETIVO DE LAS APPS UC3M	39
TABLA 3.2: FORMATO DE TABLA DE HISTORIA DE USUARIO	43
TABLA 3.3: HU-001 - SELECCIONAR OPCIONES DE CONFIGURACIÓN	44
TABLA 3.4: HU-002 - BUSCAR UN LUGAR.....	45
TABLA 3.5: HU-003 - BUSCAR UNA PERSONA.....	46
TABLA 3.6 HU-004 - VISUALIZAR AYUDA Y CRÉDITOS.....	47
TABLA 3.7: HU-005 - PROVEER LOCALIZACIÓN A APP EXTERNA	47
TABLA 3.8. FORMATO DE TABLA DE ESPECIFICACIÓN DE REQUISITO	48
TABLA 3.9: RF-001 - SELECCIÓN DEL TIPO DE BÚSQUEDA.....	49
TABLA 3.10: RF-002 - BUSCAR SEGÚN UN TÉRMINO DE BÚSQUEDA	49
TABLA 3.11: RF-003 - MOSTRAR DETALLES DE UN LUGAR	49
TABLA 3.12: RF-004 - MOSTRAR DETALLES DE UNA PERSONA.....	50
TABLA 3.13: RF-005 - REPORTAR INCIDENCIA	50
TABLA 3.14: RF-006 - ENVIAR CORREO ELECTRÓNICO A UNA PERSONA	50
TABLA 3.15: RF-007 - LLAMAR A UNA PERSONA.....	50
TABLA 3.16: RF-008 - AGREGAR A UNA PERSONA COMO CONTACTO.....	51
TABLA 3.17: RF-009 - MOSTRAR CRÉDITOS Y AYUDA	51
TABLA 3.18: RF-010 - SELECCIONAR CAMPUS	51
TABLA 3.19: RF-011 - SELECCIONAR TIPO DE MAPA	51
TABLA 3.20: RF-012 - VISUALIZAR POSICIÓN ACTUAL.....	52
TABLA 3.21: RF-013 - PROVEER SERVICIO DE LOCALIZACIÓN.....	52
TABLA 3.22: RI-001 - API DE COMUNICACIÓN	52
TABLA 3.23: RI-002 - IDIOMA DE LA APLICACIÓN.....	53
TABLA 3.24: RI-003 - VISTA PRINCIPAL.....	53
TABLA 3.25: RI-004 - VISTA DE CONFIGURACIÓN	53
TABLA 3.26: RI-005 - VISTA DE DETALLE DE LUGAR	54
TABLA 3.27: RI-006 - VISTA DE DETALLE DE PERSONA.....	54
TABLA 3.28: RI-007 - VISTA DE ACERCA DE.....	55
TABLA 3.29: RI-008 - VISTA DE CRÉDITOS.....	55
TABLA 3.30: RI-009 - VISTA DE AYUDA.....	55
TABLA 3.31: RI-010 – COMUNICACIÓN CON EL SERVICIO WEB DEL SEL-UC3M.....	56
TABLA 3.32: RR-001 – NÚMERO MÁXIMO DE RESULTADOS DE BÚSQUEDA	56
TABLA 3.33: RD-001 - VERSIÓN DEL SISTEMA OPERATIVO	56

TABLA 5.1: PLANTILLA DE PRUEBA DE SISTEMA	100
TABLA 5.2: PS-1.1 - INICIO DE LA APLICACIÓN.....	100
TABLA 5.3: PS-2.2 - COMPROBACIÓN DE TIPOS DE MAPAS	101
TABLA 5.4: PS-3.3 - PRUEBA DE SISTEMA DE BÚSQUEDA DE UN LUGAR	102
TABLA 5.5: PS-4.3 - PRUEBA DE SISTEMA DE REPORTE DE INCIDENCIA DE UN LUGAR....	103
TABLA 5.6: PS-5.4 - PRUEBA DE SISTEMA DE BÚSQUEDA DE UNA PERSONA.....	104
TABLA 5.7: PS-6.4 - PRUEBA DE SISTEMA DE LA VISTA DE DETALLE DE PERSONA	105
TABLA 5.8: PS-7.5 PRUEBA DE SISTEMA DE LA VISTA DE ACERCA DE	106
TABLA 6.1: COSTE ASOCIADO AL PERSONAL.....	122
TABLA 6.2: COSTE DE AMORTIZACIONES	123
TABLA 6.3: COSTE TOTAL.....	123

CAPÍTULO 1 INTRODUCCIÓN

1.1	PROBLEMA	2
1.2	MOTIVACIÓN DEL PROYECTO.....	3
1.3	OBJETIVOS.....	4
1.4	MÉTODO DE RESOLUCIÓN	4
1.5	TERMINOLOGÍA.....	5
1.5.1	GLOSARIO DE TÉRMINOS.....	5
1.5.2	ABREVIATURAS.....	6
1.6	CONTENIDO DE LA MEMORIA	7

En este capítulo se realiza una introducción al proyecto exponiendo cuál es el problema a resolver, cuál es su motivación, cuáles son sus objetivos y el método de resolución empleado para llevarlo a cabo. Además, se incluyen dos secciones para facilitar la lectura del presente documento en las que se definen la terminología utilizada y se explica cuál es su contenido.

1.1 PROBLEMA

El problema que se ha detectado y que se quiere resolver con este proyecto es la dificultad que tienen algunas personas para encontrar lugares que están dentro de los campus de la Universidad Carlos III de Madrid, UC3M en adelante, conociendo algún dato de los mismos.

Este problema afecta a los siguientes grupos de personas en las siguientes situaciones que son habituales encontrar en la UC3M:

- Nuevos estudiantes que al inicio de las clases, no están familiarizados con el sistema de numeración de lugares y/o no conocen el campus.
- Estudiantes de un campus que por cambio de aula puntual o en el día de un examen, no saben encontrar el aula porque no está en un edificio que conocen.
- Miembros de la comunidad universitaria que por algún motivo, tienen que visitar otro campus en el que no han estado o no conocen bien.
- Visitantes de distinta índole de la UC3M que acuden a reuniones, congresos o actividades programadas en la misma como puedan ser espectáculos culturales, eventos deportivos o actos académicos.

Los datos que estos grupos de personas conocen sobre el lugar al que quieren ir son su identificador, el edificio en el que se encuentra y/o el nombre por el que se conoce en el campus al que pertenece.

El identificador es un código numérico que indica el edificio del campus, la planta del edificio y la sala de la planta del lugar. El identificador no es homogéneo para los cuatro campus con los que cuenta la UC3M ya que para los de Leganés y Colmenarejo, se añade la zona de la planta antes de indicar la sala.

Sobre el nombre por el que se conoce un lugar en un campus, se distinguen por un lado los que por su representatividad, son ubicados fácilmente por las personas que lo conocen como por ejemplo el géiser, la locomotora y el punto de encuentro, y por otro, los que es común encontrar en un campus universitario, como la cafetería o el punto de información. Estos últimos también son ubicados fácilmente por las personas que conocen un campus pero no por las personas poco familiarizadas con el mismo.

A raíz de buscar una solución para el problema descrito, también se ha detectado que es un problema muy común querer encontrar dónde se ubica un docente dentro de un campus. Este problema se reduce a encontrar el despacho del docente y en este caso, como datos, la persona puede conocer su nombre, sus apellidos y/o su correo electrónico. Aparece principalmente porque hay personas que no son buenas recordando estructuras como el identificador del lugar. Las situaciones habituales en las que se da

son cuando un estudiante quiere ir a ver a un profesor para una tutoría u otro docente tiene que ir a visitar a un compañero, ya sea de su mismo o diferente campus.

1.2 MOTIVACIÓN DEL PROYECTO

Las principales motivaciones para realizar este proyecto son:

- **Ayudar a quienes** en algún momento de su relación con la UC3M, **deseen encontrar un lugar** de alguno de sus campus **o personas** que desempeñen su actividad profesional en ellos.
- **Acceder** de forma rápida a toda la **información relacionada** de cada uno de los **lugares de la UC3M** y **proveer una base de datos** que almacene la **georreferenciación** de todos los **lugares de la UC3M accesible** a todos los miembros de su comunidad universitaria.

Otra motivación importante es la gran expansión y popularización que en los últimos años están logrando las tecnologías móviles. Estas tecnologías permiten a los usuarios acceder a la información que necesitan de una forma cómoda y rápida sin depender del sitio en el que se encuentren, encajando estas características con las situaciones habituales en las que se detecta el problema a solucionar.

Finalmente, el interés del grupo de investigación SEL-UC3M a través del equipo de trabajo miSEL y de la UC3M por desarrollar aplicaciones que faciliten la vida universitaria a los miembros de su comunidad, además de querer mejorar la experiencia en su paso por la institución a aquellas personas que tienen una relación esporádica con ella, es otra motivación destacada para enmarcar el presente proyecto final de carrera en desarrollar una solución al problema presentado.

La disponibilidad de un espacio físico en el grupo de investigación SEL-UC3M dotado con el hardware y software necesario para desarrollar el proyecto y la cercanía y contacto que supone con el equipo de trabajo miSEL, es un aliciente para decantarse por su elección.

El propósito del proyecto, ligado a las motivaciones descritas, consiste en desarrollar las siguientes utilidades:

- Una aplicación móvil en la plataforma iOS para la provisión de servicios básicos de localización indoor en la UC3M, incluyendo búsqueda de lugares y personas en todos sus campus. También se ofrecerá información acerca del lugar o la persona buscados y la posibilidad de interactuar con ella.
- Una base de datos geográfica en la que se geolocalizan todos los lugares que se encuentran en los campus de la UC3M como aulas, despachos y otros espacios universitarios y representativos para que los usuarios tengan la posibilidad de encontrar el mayor número de lugares y personas.

- Un repositorio de planos que tiene como finalidad mostrar al usuario los planos superpuestos de una planta sobre un edificio de un campus de la UC3M. Esta característica hace que sea más sencillo identificar y llegar al lugar deseado ya que en los planos, se muestran los accesos a los pasillos y escaleras con colores para mejorar la interpretación.

1.3 OBJETIVOS

El objetivo principal de este proyecto es **desarrollar un conjunto de servicios integrados que faciliten la localización indoor de lugares a los usuarios en los diferentes campus de la Universidad Carlos III de Madrid.**

Del objetivo principal se deducen los siguientes, imprescindibles para lograr la consecución del proyecto:

- **Desarrollo de un servicio de localización para la UC3M:** en primer lugar, se deberá poner en marcha un servicio Web que permita georreferenciar lugares dentro de los campus de la UC3M. También se podrán realizar consultas georreferenciadas. Para ello, debe estar sincronizado con el directorio de PAS y PDI de la UC3M.
- **Georreferenciación de los campus de la UC3M:** el siguiente paso para realizar el proyecto consiste en georreferenciar todas las aulas, despachos, otros espacios universitarios y lugares significativos de cada uno de los campus de la UC3M.
- **Adaptación de los planos de los edificios:** complementario al trabajo de georreferenciación, es necesario adaptar los planos de las plantas de los edificios de la UC3M de manera que se puedan mostrar superpuestos en un mapa de Google.
- **Desarrollo de una aplicación iOS:** los servicios a desplegar en los objetivos anteriores han de ser utilizados en una app iOS para permitir a un usuario buscar lugares y personas, ofreciendo como resultado dónde se encuentran dentro de un campus de la UC3M e información de interés adicional asociada a la búsqueda.

1.4 MÉTODO DE RESOLUCIÓN

El método llevado a cabo para la resolución del problema ha sido iterativo puesto que en un principio, la totalidad de los requisitos que debía cumplir el sistema no estaban claros y su definición, realizada a lo largo del desarrollo del proyecto, dependía de los avances y opciones que fueran apareciendo.

Las iteraciones que se plantearon aplicables desde el comienzo del proyecto, están basadas en las que se proponen en algunas metodologías ágiles, en las que las reuniones del equipo de desarrollo toman una importancia considerable. La frecuencia de las reuniones se estableció al inicio del proyecto en dos semanas, tiempo en el que los desarrolladores implementaban la funcionalidad o funcionalidades acordadas en una reunión para revisarlas en la siguiente.

En cada una de las iteraciones, en primer lugar había una reunión con todo el equipo de desarrollo para evaluar el prototipo funcional de la iteración anterior si lo había. El prototipo reunía las funcionalidades acordadas en la reunión anterior de manera que el cliente, pudiera tener software utilizable al final de cada una de las iteraciones y definir los requisitos que el sistema debía cumplir en la reunión siguiente.

Al finalizar las reuniones, se proporcionaba a los desarrolladores una lista de tareas a realizar en un tiempo estipulado, para en la siguiente verificar cuáles se han cumplido.

Para poder realizar el seguimiento de las tareas, se ha utilizado la herramienta de gestión de proyectos *Redmine* en la que es posible crear tareas y asignárselas al desarrollador o desarrolladores que debe encargarse de realizarlas. En ella, los desarrolladores pueden indicar qué porcentaje han completado de cada tarea, indicar el número de horas empleado para ello y añadir observaciones si fuera necesario. Esta información puede ser visualizada en cualquier momento por el responsable del proyecto.

1.5 TERMINOLOGÍA

En este apartado se recoge un glosario en el que se explica brevemente algunos de los términos utilizados en el presente documento con el fin de que el lector sea capaz de comprender la totalidad de su contenido.

1.5.1 GLOSARIO DE TÉRMINOS

App: forma de denominar a las aplicaciones informáticas para dispositivos móviles como son los smartphones o las tablets.

Framework: conjunto de clases que implementan una o varias funcionalidades y que pueden ser incluidas en una aplicación para utilizar las funcionalidades que proveen.

Geolocalización o georreferenciación: concepto que hace referencia a la identificación de una situación geográfica de manera automática.

iOS: sistema operativo móvil de Apple desarrollado originalmente para su smartphone iPhone y siendo después usado en dispositivos como el iPod Touch, iPad y el Apple TV.

Smartphone o teléfono inteligente: teléfono móvil construido sobre una plataforma informática móvil, con una mayor capacidad de computación y conectividad que un teléfono móvil convencional.

Wifi: mecanismo de conexión de dispositivos electrónicos de forma inalámbrica. Generalmente se utiliza para acceder a Internet a través de un router que soporte esta tecnología.

1.5.2 ABREVIATURAS

API del inglés *Application Programming Interface*, Interfaz de Programación de Aplicaciones. Conjunto de funciones y procedimientos que ofrece una biblioteca para ser utilizada por otro software como una capa de abstracción.

GPS del inglés *Global Positioning System*, Sistema de Posicionamiento Global. Es un sistema global de navegación por satélite (GNSS) que permite determinar en todo el mundo la posición de un objeto con una precisión hasta de centímetros, aunque lo habitual son unos pocos metros de precisión.

JSON del inglés *JavaScript Object Notation*, Notación de Objetos Javascript. Es un formato ligero de intercambio de datos que no requiere el uso de XML en su notación.

LDAP del inglés *Lightweight Directory Access Protocol*, Protocolo Ligero de Acceso a Directorios. Es un protocolo que permite acceder a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

PAS Personal de Administración y Servicios.

PDI Personal Docente e Investigador.

PFC Proyecto Fin de Carrera.

SDK del inglés *Software Development Kit*, Kit de Desarrollo de Software. Conjunto de bibliotecas que ofrecen al desarrollador unas funcionalidades básicas que permiten crear programas con mayor facilidad.

SO Sistema Operativo.

UML Unified Modeling Language. Lenguaje gráfico de modelado de aplicaciones software.

XML eXtended Markup Language. Lenguaje de marcado desarrollado por el World Wide Web Consortium (W3C). Deriva del lenguaje SGML y permite definir la gramática de lenguajes específicos para estructurar documentos grandes.

1.6 CONTENIDO DE LA MEMORIA

En esta sección, se explica brevemente el contenido de cada uno de los capítulos del presente documento.

En el Capítulo 1, Introducción, se introduce el problema que se pretende resolver con el PFC, se expone la motivación que lleva a desarrollar una solución al problema, se establecen los objetivos que se quieren cumplir con la solución, se describe el método de resolución del problema, se realizan una serie de aclaraciones sobre la terminología utilizada y por último, se ofrece una visión general del contenido de la memoria.

En el Capítulo 2, Estado del Arte, se realiza un análisis de las aplicaciones móviles y web que guardan relación con los objetivos que se establecen en el Capítulo 1, se da una visión amplia sobre el entorno de desarrollo seleccionado incluyendo el estado actual de los entornos móviles y del sistema operativo iOS junto a su entorno de desarrollo. Por último, se explica el marco de la metodología utilizada para el desarrollo del proyecto y cuál ha sido la elegida.

En el Capítulo 3, Análisis de la Aplicación, se efectúa una descripción de la aplicación en su contexto para poder acometer posteriormente en el capítulo la definición de las historias de usuario y de los requisitos funcionales y no funcionales de la aplicación. En la descripción de la aplicación se ofrece principalmente una perspectiva del producto, sus funciones y la clasificación de los tipos de usuario que lo van a utilizar.

En el Capítulo 4, Diseño de la Aplicación, se describen la primera y la última versión de los *paper prototypes* utilizados para modelar la apariencia externa de la aplicación, las alternativas de diseño del sistema completo, que incluye a la aplicación y a los servicios web utilizados, que dan paso al diseño del mismo haciendo uso de los diagramas de componentes. Seguidamente se explica el diagrama de clases de la aplicación y sus diagramas de secuencia. Para cerrar el capítulo, se ofrece el diseño de la base de datos y del repositorio de planos permitiendo al usuario conocer la mayor parte del diseño del sistema.

En el Capítulo 5, Pruebas de la Aplicación, se definen las pruebas que se han realizado a la aplicación para asegurar su calidad. Se han incluido las pruebas unitarias, las de integración y las de sistema. Todas ellas se han superado correctamente.

En el Capítulo 6, Conclusiones y Líneas Futuras, se exponen las conclusiones que al autor le ha despertado la realización del proyecto y las líneas futuras sobre las que se puede trabajar para mejorar la aplicación ofreciendo nuevas funcionalidades en la línea de la mejora de la vida universitaria.

CAPÍTULO 2 ESTADO DEL ARTE

2.1 ANÁLISIS DE APLICACIONES WEB Y MÓVILES RELACIONADAS CON LOS OBJETIVOS ESTABLECIDOS	10
2.1.1 GOOGLE MAPS INDOORS	11
2.1.2 POINT INSIDE SHOPPING & TRAVEL.....	12
2.1.3 MYCOEX.....	13
2.1.4 MALL RAT	13
2.1.5 WIFARER.....	14
2.1.6 UNIVERSIDAD DE GRANADA	15
2.1.7 UNIVERSITY OF ALABAMA	15
2.1.8 MIT MOBILE	16
2.2 ENTORNO DE DESARROLLO MÓVIL SELECCIONADO	20
2.2.1 ENTORNOS MÓVILES.....	20
2.2.2 VISIÓN GENERAL SOBRE IOS	21
2.2.3 ESTRUCTURA DEL SISTEMA IOS.....	24
2.2.4 CICLO DE VIDA DE LAS APPS IOS	25
2.2.5 ENTORNO DE DESARROLLO INTEGRADO.....	27
2.2.6 OBJECTIVE-C.....	28
2.3 DESARROLLO ÁGIL COMO MÉTODO PARA REALIZAR EL PROYECTO	28
2.3.1 THE AGILE ALLIANCE.....	29
2.3.2 SCRUM.....	30
2.4 CONCLUSIONES DEL ANÁLISIS.....	32

Antes de comenzar a desarrollar un proyecto en el que se va a invertir tiempo y recursos, es necesario conocer el estado actual del problema que se pretende resolver analizando el contexto en el que se enmarca y las aplicaciones existentes, si las hubiera, que ofrezcan una solución al mismo o a similares que se hayan planteado. De este modo, se obtiene una visión general de la que se concluirá si es necesaria una nueva aportación en el campo de aplicación.

El estudio de las aplicaciones existentes está motivado en que el propósito no es partir desde cero, sino que lo que se quiere es adoptar aspectos conocidos de las mismas que son adecuados para la resolución del problema y suplir las carencias que se detecten, mejorando los que tienen baja madurez e incorporando los que se necesiten. La solución a proponer se ajustará por tanto al problema que se pretende resolver.

Como contenido del estado del arte, en este capítulo se expone la información obtenida al investigar acerca de los temas que se tratarán a lo largo del desarrollo del proyecto. Para ello, se expone en primer lugar un análisis de aplicaciones móviles y web relacionadas con los objetivos establecidos. A continuación se explica cuál es el entorno de desarrollo elegido y cuáles son sus características principales. Por último se explica la forma en la que se ha utilizado el desarrollo ágil como método para realizar el proyecto.

2.1 ANÁLISIS DE APLICACIONES WEB Y MÓVILES RELACIONADAS CON LOS OBJETIVOS ESTABLECIDOS

Para definir el estudio a realizar sobre las aplicaciones existentes, es importante tener claros los objetivos del presente PFC, establecidos en la sección 1.3, de los que hay que recordar brevemente en esta sección dos principales:

- Crear una base de datos que almacene la georreferenciación de todas las aulas, despachos, otros espacios universitarios y lugares significativos de cada uno de los campus de la UC3M.
- Desarrollar una aplicación iOS que integre los servicios desplegados para la búsqueda georreferenciada de lugares y personas de la UC3M.

Teniendo en cuenta estos objetivos, el estudio se ha iniciado a partir de dos puntos. El primero, es el artículo de Alonso, García y Amescua [1] y el segundo, la búsqueda de aplicaciones y soluciones desde el buscador web Google y desde el buscador de la tienda de aplicaciones de Apple AppStore.

En el artículo referido, se hace una investigación acerca del aprendizaje a través del móvil y de las aplicaciones móviles desarrolladas para las universidades españolas. Indica que sólo el 17,94% de las universidades españolas se han sumergido en el ecosistema de las aplicaciones móviles. Señala que las apps analizadas más estables son las desarrolladas para iOS frente a las desarrolladas para Android, que son las que más veces fallan, hecho que atribuye al control poco estricto que Google realiza antes de su publicación. El artículo concluye mencionando que las aplicaciones universitarias encontradas son meramente informativas y no son demasiado complejas y además, no explotan en gran medida la posibilidad de desarrollar apps que faciliten la vida universitaria.

Sobre las búsquedas, como se ha indicado, se han realizado en el buscador web Google y en el buscador de la aplicación AppStore, disponible en los dispositivos con iOS y por descarga directa en diferentes plataformas.

El criterio para la definición de las búsquedas ha sido diferente para cada buscador y en ambos, se han introducido tanto en inglés como en castellano.

Las búsquedas en el buscador web Google se han orientado a aplicaciones o soluciones que permiten una localización en interiores más precisa que la ofrecida por redes móviles y GPS. Los términos de búsqueda utilizados han sido:

- wifi indoor position maps
- indoor location solution

Por el contrario, en la AppStore las búsquedas se han orientado a encontrar aplicaciones móviles universitarias y comerciales que hagan uso de mapas y geolocalización. Los términos de búsqueda utilizados han sido:

- universidad mapas
- university maps
- maps
- indoor
- navigation

Se elaboró una primera búsqueda al inicio del proyecto durante los primeros días del mes de noviembre de 2011 y otra a mediados del mes de abril de 2012 para comprobar las novedades que fueran apareciendo.

Como resultado de las búsquedas con los términos definidos para los buscadores utilizados y de las aplicaciones analizadas en el artículo de investigación, se ha obtenido una lista de aplicaciones. En los siguientes apartados, se describen las más importantes por contener funcionalidades relevantes.

2.1.1 GOOGLE MAPS INDOORS

Google ofrece un servicio de mapas cartográficos, Google Maps, cuyo punto fuerte es la búsqueda de calles en todos los países del mundo que le han dado permiso para tomar capturas vía satélite de su territorio. Este servicio ha alcanzando un nivel de calidad importante porque proporciona perspectiva 2D de los planos y permite establecer rutas entre diferentes puntos geográficos. El nivel de madurez también es alto puesto que la mejora del servicio ha sido constante. Todo esto ha llevado a Google Maps a ser ampliamente utilizada por la comunidad de usuarios.

A finales de noviembre de 2011 apareció en la versión móvil de Google Maps [2], sólo disponible en dispositivos Android, una nueva funcionalidad denominada *Indoors* que permite la visualización de interiores y el posicionamiento en los mismos. Sobre el mapa cartográfico se muestra un plano esquemático de la planta de un edificio para el que está disponible la funcionalidad. En el plano se muestran servicios que hay en la planta del edificio. Los edificios pertenecen a centros comerciales o aeropuertos entre otros. Para más información sobre *Indoors*, consultar [3].

Resulta por tanto de interés la navegación para llegar de una calle a otra ofreciendo una o varias rutas y la evolución que puede tomar la funcionalidad *Indoors* ya que está en línea con uno de los objetivos del desarrollo del proyecto.

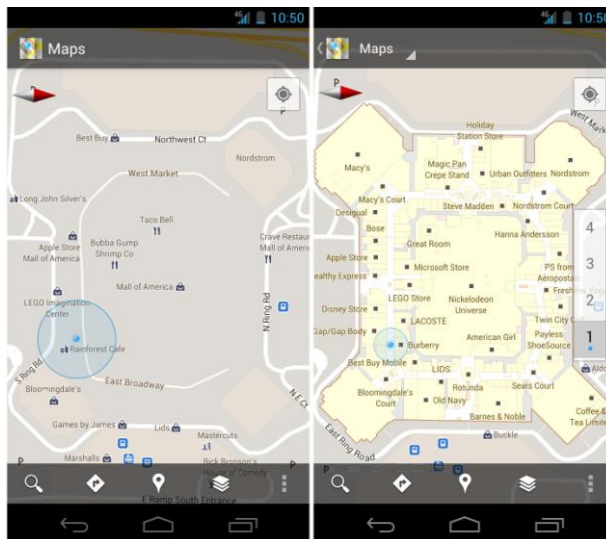


Figura 2.1: Google Maps (Android)

2.1.2 POINT INSIDE SHOPPING & TRAVEL

La aplicación Point Inside [4] permite visualizar diferentes tipos de servicios disponibles en edificios de entidades públicas o privadas como aeropuertos o centros comerciales situados por todo el mundo. Se proporciona un plano esquemático dibujado de cada planta de cada uno de los edificios y para cada planta es posible seleccionar los servicios que se desean visualizar de entre los disponibles como restaurantes, aseos o tiendas. Además, en el caso de lugares como un aeropuerto que se componen de varios edificios, permite elegir la planta a mostrar de un edificio en concreto para que el usuario preste atención únicamente al edificio en el que se está buscando.

El punto fuerte de esta aplicación es la facilidad de uso para cambiar entre las plantas de un edificio y los puntos que se desean visualizar. Los planos de las plantas, al ser esquemáticos y al poder seleccionar los servicios a visualizar, no distraen al usuario con otros elementos. También destacar el amplio repositorio de planos del que dispone, el cual es accesible sin conexión a Internet.

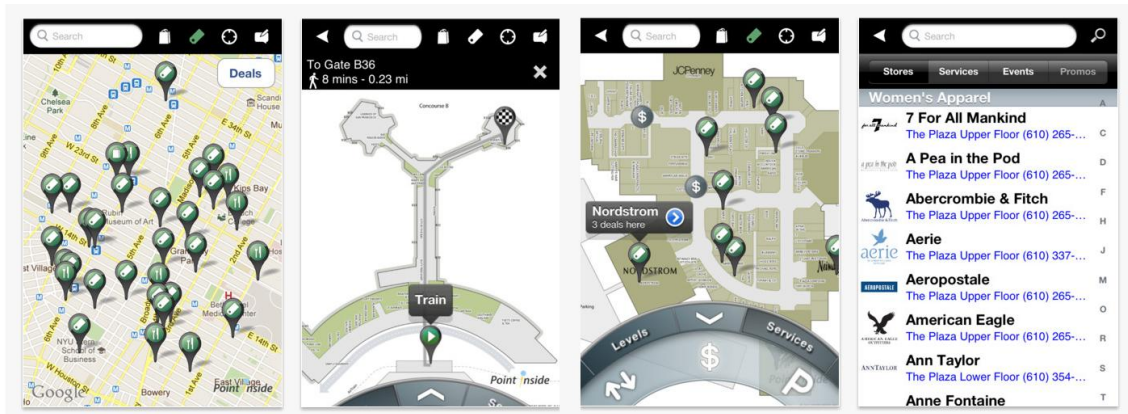


Figura 2.2: App Point Inside & Travel

2.1.3 MYCOEX

La app myCoex [5] ofrece información de ferias, eventos y tiendas localizadas en Coex, el mayor centro comercial bajo tierra de Asia situado en Corea del Sur. Permite descarga cupones de descuento y el registro en actividades y eventos. El punto fuerte de la aplicación es que permite la visualización de planos esquemáticos dibujados mostrando servicios y además crea una ruta entre los servicios que el usuario elija. Además incluye otras funcionalidades que giran en torno a Coex.

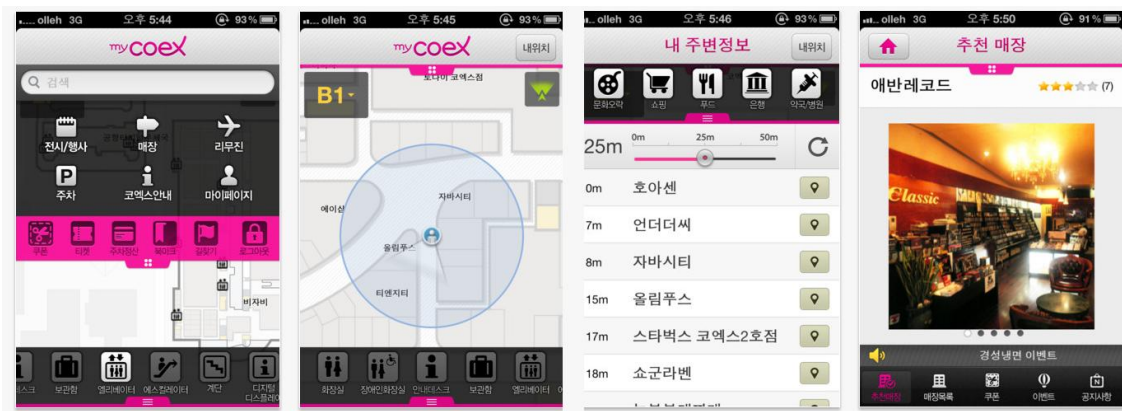


Figura 2.3: App MyCoex

2.1.4 MALL RAT

Mall Rat [6] es una aplicación similar a MyCoex y Point Inside Shopping & Travel especializada en ofrecer a los usuario planos de centros comerciales de todo el mundo. Como funcionalidades más destacadas permite visualizar la información de una tienda cercana al usuario o seleccionándola de una lista, recuerda las búsquedas recientes, realizar planes de compra en las tiendas de una planta y permite la navegación indoor mostrando la ruta desde la tienda en la que se encuentra el usuario hasta la que quiere llegar.

Como el resto de apps de centros comerciales analizadas, muestra de forma esquemática los planos dibujados de cada planta y están disponibles cuando el dispositivo no tiene conexión a Internet. La interfaz de usuario, aunque es intuitiva, puede ser mejorable.

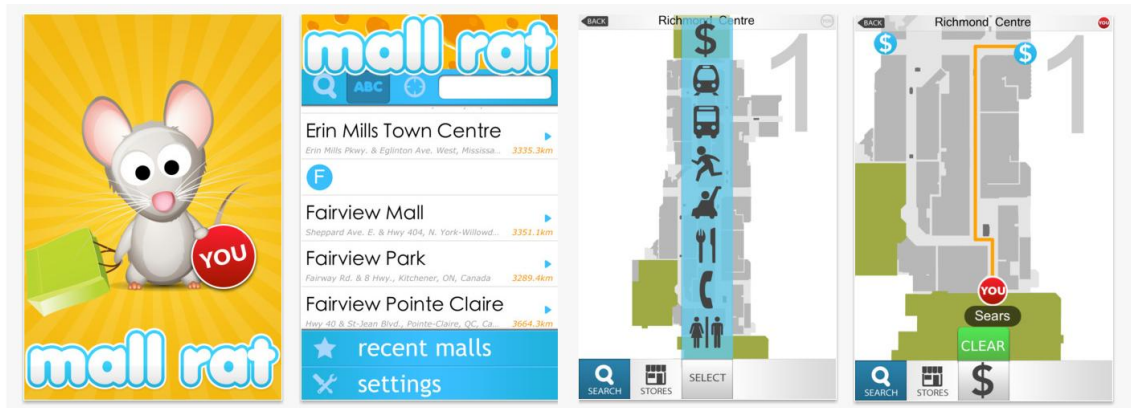


Figura 2.4: App Mall Rat

2.1.5 WIFARER

La aplicación Wifarer [7] provee posicionamiento y navegación en interiores de locales y centros asociados en Estados Unidos y Canadá. El punto fuerte de la aplicación es que en todo momento conoce el punto exacto en donde el usuario está situado basándose en la tecnología WiFi. Ofrece un directorio de los lugares de cada centro y es capaz de trazar una ruta entre dos lugares del centro. Además ofrece información que puede interesar al usuario de los lugares del centro más cercanos a su posición. Incluye un mapa esquemático dibujado de cada una de las plantas de un centro.

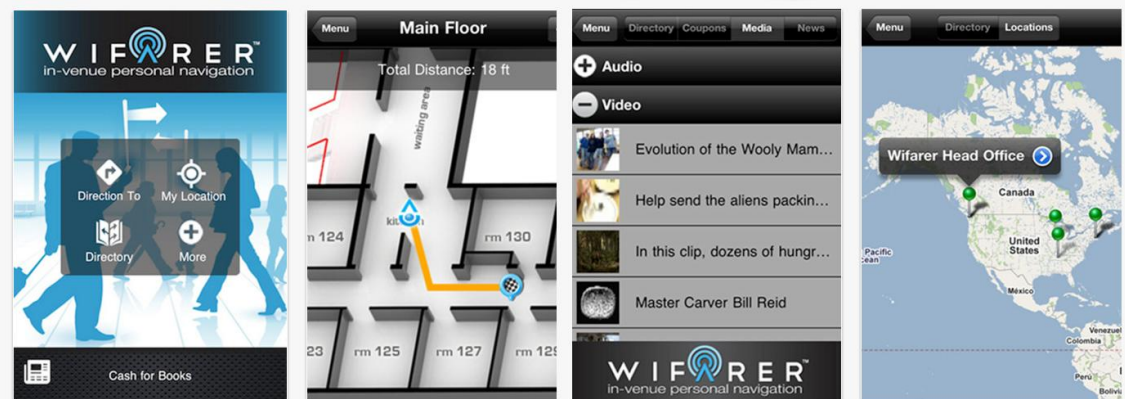


Figura 2.5: App Wifarer

2.1.6 UNIVERSIDAD DE GRANADA

La Universidad de Granada ofrece una suite de aplicaciones corporativas contenidas en una principal [8]. Entre las disponibles se encuentra una que permite visualizar cada uno de los edificios de la citada universidad señalándolos en un mapa.



Figura 2.6: App de la Universidad de Granada

2.1.7 UNIVERSITY OF ALABAMA

La Universidad de Alabama ofrece una suite de aplicaciones corporativas contenidas en una principal [9] al igual que la Universidad de Granada. Entre las disponibles también se encuentra una que permite visualizar los edificios de la universidad pero a diferencia de la de la Universidad de Granada, permite buscar los edificios por separado en lugar de mostrar todos directamente. Utiliza un plano esquemático dibujado para enseñar los edificios superpuesto en un servicio de mapas cartográfico y da la posibilidad de visualizar más información del edificio.

Por otro lado, incluye una aplicación que muestra el directorio de personal de la universidad ofreciendo información de contacto de la persona buscada.

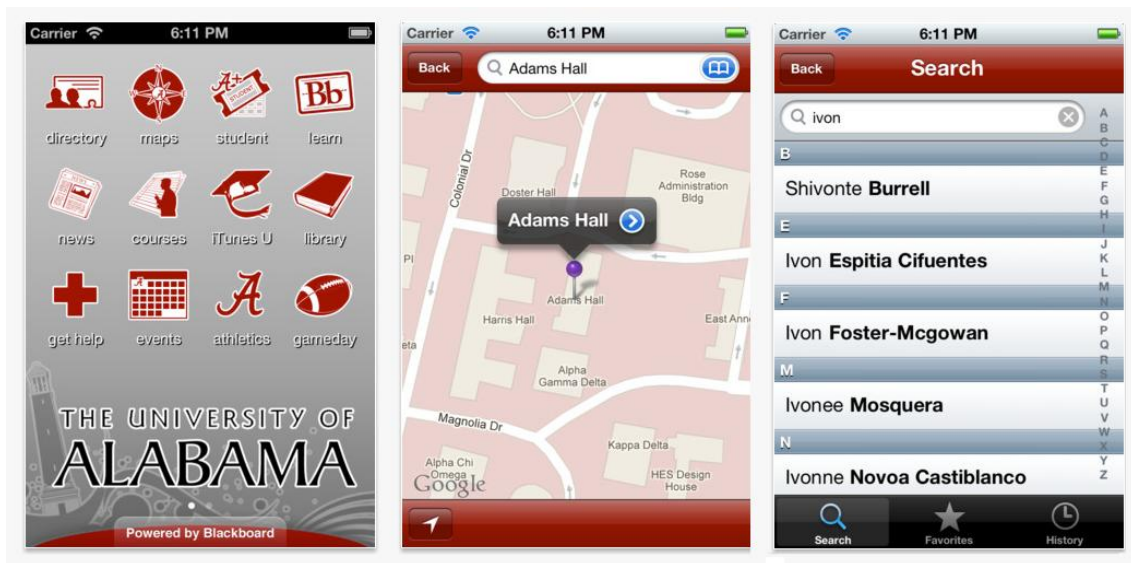


Figura 2.7: App de la Universidad de Alabama

2.1.8 MIT MOBILE

El Massachusetts Institute of Technology, MIT, ofrece una aplicación corporativa [10] que contiene una aplicación para la búsqueda de edificios del campus. Muestra un plano esquemático de todos los edificios del campus sobre un servicio de mapas cartográficos y ubica el edificio buscado en el campus posibilitando conocer más información del mismo.

Por otro lado, la app del MIT ofrece una aplicación de directorio que permite buscar todo el personal docente, administrativo y estudiante del MIT con la posibilidad de enlazar directamente con la aplicación de mapas para ver el edificio en el que se encuentra el personal docente y administrativo buscado. Además, proporciona información de contacto de la persona buscada. La conexión con la aplicación de mapas también la realiza la aplicación de eventos para mostrar el edificio de celebración del evento.

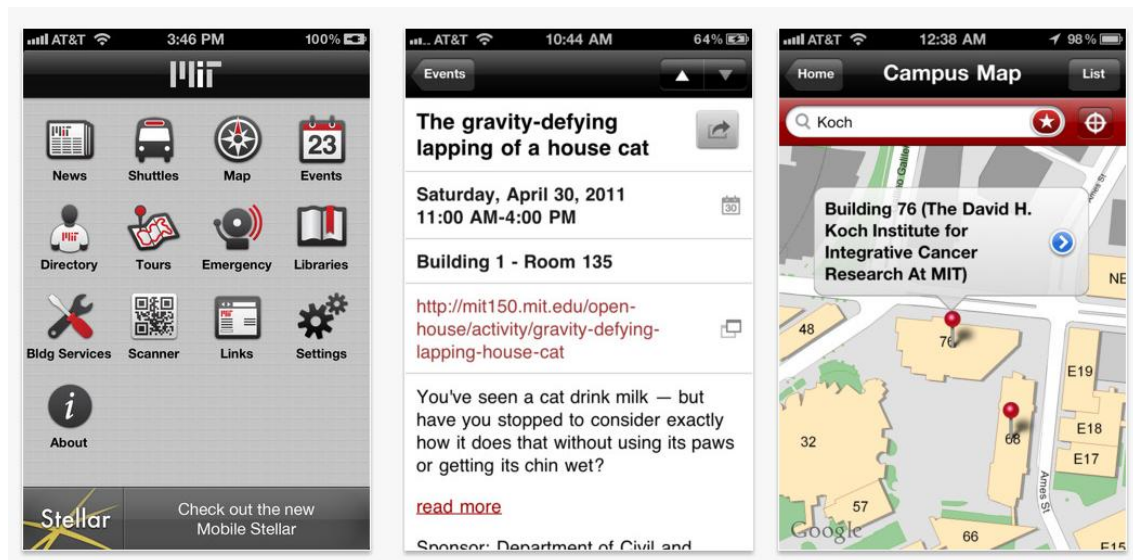


Figura 2.8: App del MIT

Después de explicar las características de cada una de las aplicaciones encontradas, se realiza un análisis comparativo entre ellas teniendo en cuenta las características más destacables de cada una de ellas. El resultado de la comparación se puede consultar en la Tabla 2.1.

	Google Maps Indoors	Point Inside	MyCoex	MallRat	Wifarer	Universidad de Granada	Universidad de Alabama	MIT Mobile
Uso de mapa cartográfico	X	X			X	X	X	X
Indicación sobre mapa de centros/edificios disponibles		X				X		
Directorio/Búsqueda de centros/edificios disponibles	X	X		X	X		X	X
Uso de plano esquemático dibujado interior	X	X	X	X	X			

Posicionamiento en interiores	X		X						
Visualización de servicios interiores	X	X	X	X	X				
Selección por tipo de servicios a visualizar		X		X					
Selección por cercanía de servicios a visualizar			X		X				
Ruta en interior entre posición y servicio			X	X	X				
Visualización de detalle de servicio/edificio				X	X		X		
Uso de plano de esquemático de campus							X	X	
Provisión de servicio de localización a otras apps									X

Tabla 2.1: Comparativa entre características de las aplicaciones

La comparativa realizada en la Tabla 2.1 permite visualizar rápidamente las características comunes a todas las aplicaciones analizadas que proveen diferentes servicios en interiores y por tanto, se debe considerar su incorporación en la aplicación enmarcada en este proyecto.

La característica que implementan todas las aplicaciones es ofrecer un sistema para dar a conocer al usuario los centros o edificios disponibles. Sólo una indica directamente en un mapa cartográfico todos los centros disponibles y el resto utiliza una búsqueda o provee un directorio para indicar los centros que se pueden visualizar. Es imprescindible implementar una funcionalidad que dé a conocer al usuario los edificios disponibles.

La característica que la mayoría de las aplicaciones implementan es hacer uso de un servicio de mapas cartográficos siendo su uso indicar la ubicación de un centro o un edificio, superponer al mapa los planos de los centros y los edificios y superponer el campus de una universidad en el mismo. Resulta recomendable utilizar servicios de este tipo para disponer de una base sobre la que comenzar a trabajar y poder aumentar la funcionalidad de la app en el futuro.

Una característica también común y que se puede incorporar es el uso de planos esquemáticos dibujados para mostrar el interior de un centro. Tienen la ventaja frente a los planos arquitectónicos que son más amigables hacia el usuario y están orientados a que, con un diseño minimalista, el usuario conozca rápidamente el entorno donde se va a mover.

Otra característica que se puede tomar es la visualización de servicios disponibles que, adaptado a una universidad sería poder ver los aseos, las escaleras o si hay algún tipo de asociación o servicio universitario alrededor. Para seleccionar el tipo de servicio a visualizar en el plano, las aplicaciones han optado por mostrar de una vez todos los servicios o permitir seleccionar los servicios por tipo o por cercanía.

Una característica interesante es el posicionamiento del usuario en interiores. Actualmente, la tecnología GPS comete un fallo considerable para posicionar a una persona en un edificio y se están desarrollando otras técnicas como el posicionamiento por WiFi en el caso de las aplicaciones analizadas, que reducen el error producido por la tecnología GPS.

Finalmente, una característica que implementa únicamente una app es la provisión de servicios de localización a otras apps. Este hecho se da en una aplicación corporativa de una universidad y proporciona una búsqueda ágil y más natural de una localización. Es una característica que resulta de utilidad y su implementación debería incorporarse al proyecto.

Tras conocer las características obtenidas de cada una de las apps y saber para que se utilizan, se ha considerado recomendable incorporar las siguientes a la aplicación que se enmarca en el proyecto:

- Uso de mapa cartográfico
- Directorio/Búsqueda de edificios disponibles
- Uso de plano esquemático dibujado interior
- Posicionamiento en interiores
- Visualización de servicios disponibles
- Ruta en interior entre posición y servicio

El resto de características se consideran deseables y de entre ellas se pueden destacar la visualización de detalle de servicio/edificio y la provisión de servicios de localización a otras apps.

2.2 ENTORNO DE DESARROLLO MÓVIL SELECCIONADO

En esta sección se analiza en primer lugar el estado de los diferentes entornos móviles, después se da una visión general sobre el sistema operativo iOS, su evolución y se explica la estructura interna del sistema. Finalmente, se introduce cuál es el entorno de desarrollo integrado (IDE) utilizado y el porqué de su elección.

2.2.1 ENTORNOS MÓVILES

Tal como se puede extraer del post publicado en la Comunidad Movistar sobre las aplicaciones más descargadas en 2012 [11] basado en el informe App Date [12] del primer trimestre del año 2012, los entornos móviles más comunes en España durante los primeros meses del año 2012, ordenados por uso, son Symbian, Android, iOS, BlackBerry y Windows Phone.

En la Figura 2.9 se observa que el uso de iOS está en torno al 22%. El hecho de que Symbian sea el de mayor volumen, actualmente puede llamar la atención pero hay que tener en cuenta el número de teléfonos móviles de propósito general, no considerados smartphones, que a día de hoy lo incorporan.

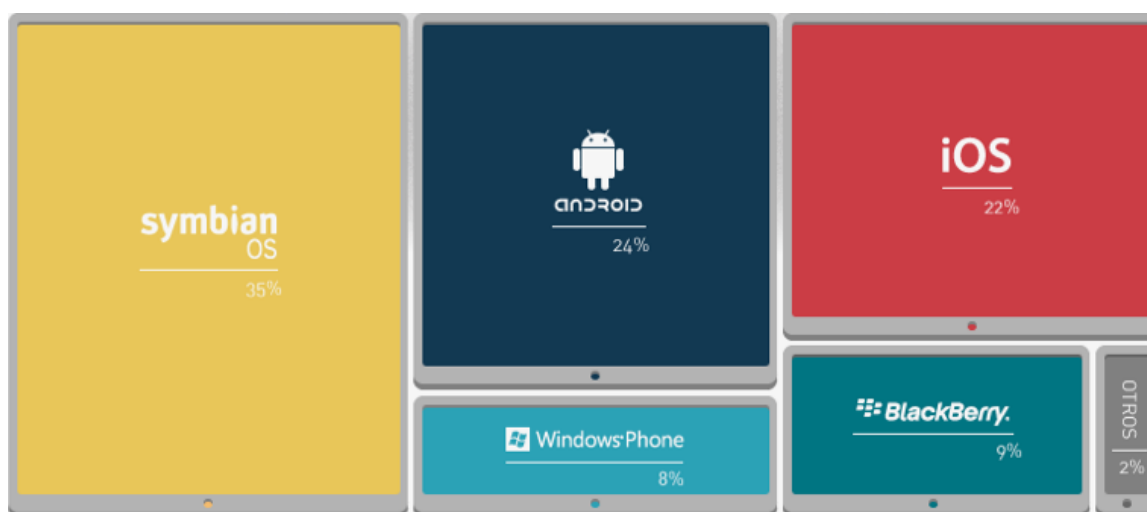


Figura 2.9: División del mercado de S.O. móviles

El sistema operativo Android, incorporado en smartphones de diferentes compañías, ocupa un 24% del total, posicionándose como un claro competidor para los dispositivos de Apple que utilizan iOS.

En el post de la Comunidad Movistar también se incluyen datos sobre el número de aplicaciones descargadas en los dispositivos que hacen uso de iOS o de Android. El número total de descargas de apps en España fue de diez millones, habiéndose descargado el conjunto de los usuarios españoles doscientas veinte mil apps en dispositivos Android y trescientas ochenta nueve mil apps en smartphones iPhone, los cuales sólo son una parte de los dispositivos que funcionan sobre iOS.

En un artículo de diciembre de 2011 publicado en la web Celularis en el que se analiza el número de apps descargadas en el mes de diciembre de 2011 en dispositivos iOS y Android [13], treinta y dos millones de apps fueron descargadas en dispositivos Android frente a los treinta y un millones en iPhone. No es así en el cómputo mundial, resultando iOS vencedor en número de descargas frente al resto de sistemas operativos para smartphones.

Por otro lado, en el artículo de la web de Celularis analiza el ingreso obtenido de la venta de apps para iOS y para Android y se indica que mientras en la plataforma Android el 0,31% de las apps que se han descargado son de pago, en iOS el porcentaje es del 19,55%.

Por tanto, los motivos que han llevado realizar el desarrollo de la solución para la plataforma iOS frente a la plataforma Android han sido:

- La aceptación de las apps para iOS tanto a nivel de España como a nivel mundial, habida cuenta del número de descargas de apps que se han registrado para cada plataforma.
- La posibilidad de ingresar dinero de forma segura por la descarga de la app. Aunque la app a desarrollar va a ser gratuita, resulta útil de cara al futuro profesional conocer más a fondo la plataforma genera más beneficios.
- Adicionalmente, la existencia del desarrollo en paralelo del mismo proyecto para la plataforma Android.

2.2.2 VISIÓN GENERAL SOBRE IOS

La aparición del sistema operativo iOS surgió en el año 2007 con el lanzamiento al mercado del smartphone iPhone de la compañía Apple. La primera generación del terminal iPhone incorporaba como sistema operativo iOS en su versión 1.0. Apple ha ido incorporando progresivamente iOS en los terminales móviles que ha producido, como es el caso de los dispositivos iPod Touch en el mismo 2007 y iPad en el año 2010, debutando con las versiones 1.1 y 3.2 de iOS respectivamente.

Las versiones de iOS se numeran de forma tradicional, siendo el número más a la izquierda el número de versión y los siguientes, separados por puntos, correspondientes al número de revisión. Cada una de las versiones no tiene un nombre característico como sucede con las del sistema operativo Android, sino que se las conoce por iOSX, representando X el número de versión.

Para acercar la evolución del sistema operativo iOS [14], en la Tabla 2.2 se listan las versiones más importantes especificando los dispositivos a los que da soporte y su fecha de lanzamiento.

Versión	Dispositivos	Fecha de lanzamiento
1.0	iPhone	29/06/2007
1.1	iPhone iPod Touch	14/09/2007
2.0	iPhone 3G iPodTouch	11/07/2008
2.1	iPod Touch y iPodTouch segunda generación	09/09/2008
3.0	iPhone 3GS iPod Touch segunda generación Mejoras limitadas para dispositivos de primera generación	17/06/2009
3.1.3	Última actualización para dispositivos iPhone y iPod Touch de primera generación	02/02/2010
3.2	iPad	03/04/2010
4.0	iPhone 4 Mejoras limitadas para iPhone 3G y iPod segunda generación	21/06/2010
4.2.1	iPhone 3G, iPhone 3GS y iPhone 4 iPodTouch tercera generación iPad Última actualización para iPod Touch segunda generación	22/10/2010
4.3	iPhone 3GS y iPhone 4 iPod Touch tercera y cuarta generación iPad y iPad2	09/03/2011
5.0	iPhone 3GS, iPhone 4 y iPhone 4S iPod Touch tercera y cuarta generación iPad, iPad 2	12/10/2011

5.1	iPad 2 y iPad 3	07/05/2012
5.1.1	Última actualización para dispositivos para iPad y iPod Touch tercera generación	07/05/2012
6.0	iPhone 5 iPod Touch quinta generación iPad 2, iPad 3, iPad 4 (23/10/2012) iPad mini (23/10/2012) Mejoras limitadas para iPhone 3GS, iPhone 4, iPod Touch cuarta generación y iPad2	19/09/2012

Tabla 2.2: Evolución del sistema operativo iOS

Además de utilizar iOS en los terminales móviles, a partir de la segunda generación del receptor digital multimedia Apple TV, Apple comenzó a incluir iOS como sistema operativo. Las actualizaciones de iOS incluidas para Apple TV más destacadas han sido las que se muestran en la Tabla 2.3.

Versión	Fecha de lanzamiento
4.1	04/09/2010
4.3	09/05/2010
5.0	12/10/2011
6.0	24/09/2012

Tabla 2.3: Evolución del sistema operativo iOS en Apple TV

Como se puede extraer de las tablas Tabla 2.2 y Tabla 2.3, iOS está desplegado en una gama de dispositivos bastante amplia, que varía entre *smartphones*, reproductores táctiles de música y tabletas para los que es posible desarrollar una app compatible con todos ellos.

Cada una de las actualizaciones destacadas ha aportado mejoras de innovación que hacen aprovechar al máximo las capacidades de los dispositivos para los que se distribuyen, abriendo a los desarrolladores un gran abanico de posibilidades para incluir y combinar, resultando en apps con gran potencial en cada contexto para el que estén diseñadas.

2.2.3 ESTRUCTURA DEL SISTEMA iOS

En el manual iPhone Application Development in 24 hours [15], se explica con detalle cada una de las capas de tecnología que componen el sistema operativo iOS mostradas en la Figura 2.10. En este apartado, se hace una referencia a cada una de ellas explicando su funcionalidad dentro del sistema operativo.

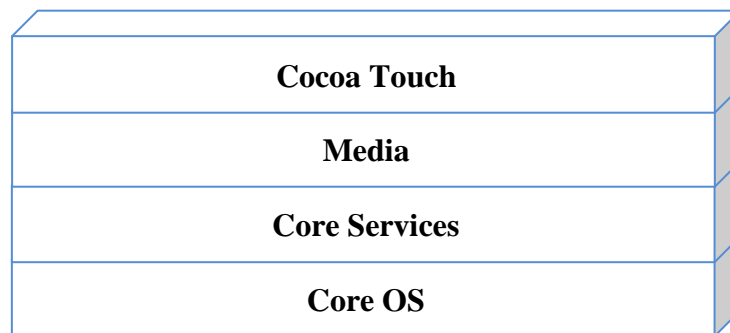


Figura 2.10: Capas de tecnología que componen el sistema iOS

Cocoa Touch

Esta capa se encuentra en el nivel más alto y permite la interacción del usuario con el dispositivo, manejando los elementos que provocan entrada y salida de datos desde y hacia el usuario. Se puede decir que provee el núcleo de funcionalidades a las apps que se desarrollan. Cabe destacar en esta capa el *framework UIKit* ya que contiene recursos más allá del manejo de interfaces.

Media

Esta capa está compuesta por los *frameworks* que proveen el manejo de creación de gráficos complejos, reproducción de audio y vídeo y la generación de gráficos 3D en tiempo real.

Core Services

Esta capa está compuesta por los *frameworks* que proveen el acceso a servicios de bajo nivel del sistema operativo. Una muestra de ellos es el acceso a ficheros, a la conexión de red y los tipos de objetos. En esta capa cabe destacar el *framework Foundation* ya que provee las funcionalidades más usadas de esta capa.

Core OS

Esta capa está compuesta por los *frameworks* que proveen el acceso a los servicios de más bajo nivel del sistema operativo como por ejemplo el manejo de hilos, el uso de matemática compleja, el acceso a los accesorios de los dispositivos y a la criptografía. El uso de estas funcionalidades no es necesario para el desarrollo de apps de propósito general.

Los *frameworks* de los que se hace uso en la mayoría de las apps son *UIKit* de la capa Cocoa Touch, *CoreGraphics* de la capa Media, y *Foundation* de la capa Core Services ya que cubren la mayoría de funcionalidades que un desarrollador incorpora en sus aplicaciones.

Hay que tener en cuenta que para el desarrollo de iOS, Apple partió para la capa Cocoa Touch de Cocoa, hecho que otorga una gran madurez a Cocoa Touch. Cocoa nació a mediados de la década de 1980 en la plataforma NeXSTEP y en 1996, Apple adquirió NeXT Computer. En los siguientes años, Cocoa pasó a ser el estándar de facto para el desarrollo de aplicaciones para Macintosh.

La ventaja de adaptar Cocoa para iOS es doble: la madurez que ofrece frente a otros sistemas operativos móviles y la facilidad de los desarrolladores para Macintosh para sumergirse en el desarrollo de apps para iOS debido a su similitud y utilización de los mismos patrones de desarrollo en ambas plataformas.

2.2.4 CICLO DE VIDA DE LAS APPS IOS

Para ilustrar el ciclo de ejecución normal de una app desarrollada para iOS, se utiliza la Figura 2.11 extraída de [15].

Como se ha indicado en el apartado 2.2.3, Estructura del sistema iOS, el *framework UIKit* provee entre otras, las funcionalidades de gestión de la interfaz de usuario y de los eventos, así como la totalidad de la gestión de la ejecución de una app. De esta forma, cuando el usuario inicia una app, el *framework UIKit* gestiona su inicialización a través de las funciones *main()* y *UIApplicationMain()*. Si es necesario, el programador ha indicado en la parte de código las acciones a llevar a cabo cuando la app es inicializada. Las acciones están definidas en el método *application:didFinishLaunchingWithOptions*.

Una vez se ha inicializado la app, entra en un bucle que recibe los eventos que se producen en ella. Un ejemplo de evento es la pulsación de un botón de la interfaz por parte del usuario y como para cualquier evento, su gestión estará definida por el programador en los métodos que están destinados a manejarlo.

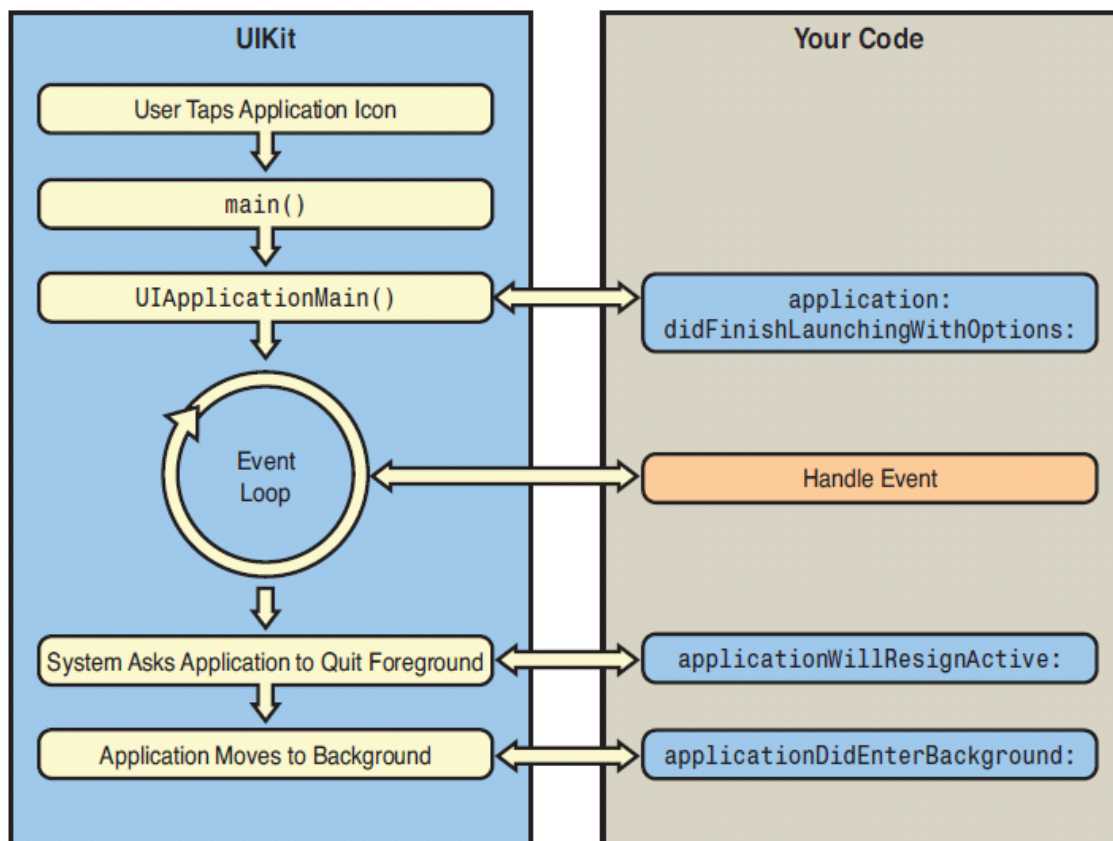


Figura 2.11: Ciclo de vida de una app iOS

El bucle de ejecución continúa hasta que se solicita a la aplicación que pase a *background*. Normalmente sucede cuando el usuario pulsa el botón Inicio del dispositivo. Cuando una app entra en *background*, su ejecución no termina y cuando el usuario la selecciona desde la pantalla de inicio o desde el gestor de tareas, vuelve al modo *foreground* en el mismo punto de ejecución que estaba antes de pasar a *background*. El método `applicationDidEnterBackground` posibilita almacenar la información necesaria para que una app siga su ejecución al volver del modo *background* y se distingue del método `applicationWillResignActive` en que este último se usa para aplicaciones que no soportan el modo *background*.

En resumen, el ciclo de ejecución de una app iOS comienza cuando el usuario la selecciona en la pantalla de inicio. Por defecto, existen unas funciones que inicializan la app y un método en el que el programador decide que acciones llevar a cabo en su inicialización. Cuando la app se inicializa, entra en un bucle del que sale cuando pasa a modo *background*. Finalmente, mientras permanece en el bucle, recibe eventos que son gestionados en los métodos destinados a manejarlo y que han sido escritos por el programador.

2.2.5 ENTORNO DE DESARROLLO INTEGRADO

El entorno de desarrollo integrado para la programación del proyecto ha sido la herramienta XCode ya que es la única que permite crear aplicaciones para los sistemas operativos Mac OS y iOS.

XCode ha ido evolucionando desde su aparición hasta convertirse actualmente en un potente entorno de desarrollo. Integra los *frameworks* Cocoa y Cocoa Touch, el Xcode IDE y herramientas de análisis. Únicamente puede instalarse en computadores con Mac OS. A continuación, se presentan las partes principales de Xcode [17]:

Xcode IDE

Permite la construcción del código fuente de las aplicaciones, el diseño de las interfaces de usuario junto con el establecimiento de sus relaciones con el código fuente y por último, la depuración del código, todo ello desde una misma ventana.

Ofrece ayuda a la edición de código, ejecutar las aplicaciones para su testeo tanto en el simulador como en un dispositivo físico y gestionar las licencias para firmar las aplicaciones que permiten el testeo y la posterior subida a la App Store.

Compilador Apple LLVM

Utiliza un mismo parseador para compilar código C/C++ y código Objective-C que es el utilizado para desarrollar aplicaciones para iOS. El compilador LLVM está integrado en el entorno además para identificar lo que el programador está escribiendo y ofrecer terminaciones de código e identificar los errores de sintaxis que comete.

Instruments

Es una herramienta para el análisis del rendimiento y comportamiento de las aplicaciones. Su uso está orientado a mejorar la experiencia del usuario y recoge datos con respecto al uso de memoria o de de CPU en tiempo real. Los datos recogidos los muestra de forma gráfica y los relaciona con líneas de código para poder identificar posibles mejoras de la aplicación.

Simulador iOS

Pensado para ahorrar tiempo a la hora de testear las apps, posibilita la simulación del comportamiento de la app si estuviera ejecutándose en un dispositivo móvil con iOS. Permite elegir la versión de iOS y el dispositivo a la hora de realizar la ejecución de la simulación. Tiene limitaciones en cuanto a que no es posible simular algunos elementos físicos como la cámara de fotos.

El desarrollo de este proyecto se comenzó utilizando la versión 4.0 de Xcode, la primera en integrar la construcción de código y diseño de interfaces en una misma ventana. La última versión de Xcode en la que se ha trabajado en el desarrollo de la app ha sido la 4.3, pudiendo comprobar su correcta ejecución hasta iOS 5.1.

2.2.6 OBJECTIVE-C

Objective-C es el lenguaje de programación utilizado para programar aplicaciones para los sistemas operativos MAC OS X y iOS de Apple.

Según se puede leer en la documentación de Apple [18], Objective-C es un lenguaje de programación orientado a objetos y de alto nivel. Surge a principios de la década de 1980 como una extensión de la estandarización ANSI del lenguaje C y está basado en Smalltalk, uno de los primeros lenguajes de programación orientados a objetos.

Se distinguen dos tipos de ficheros, los de interfaz, con extensión *.h*, y los de implementación, con extensión *.m*. En los primeros se encuentran las definiciones de los atributos y métodos de la clase mientras que en los segundos, se implementan los métodos y se hace uso de los atributos definidos.

Se incluye un tipo de fichero que permite construir la interfaz de usuario. Este fichero tiene la extensión *.xib* y se le conoce como fichero NIB. Para poder acceder a los elementos de la interfaz de usuario, se definen en los ficheros de interfaz una serie de conectores correspondientes a los diferentes elementos de la interfaz de usuario. En el fichero NIB se establece la relación entre cada conector y cada elemento y en el fichero de implementación, se accede a los elementos.

Una característica diferenciadora de Objective-C es que es un lenguaje de programación descriptivo. La definición de los métodos está enfocada a conseguir explicar en su nombre qué es lo que hace el método. Haciendo uso de esta característica, se consigue escribir un código fácil de interpretar.

2.3 DESARROLLO ÁGIL COMO MÉTODO PARA REALIZAR EL PROYECTO

Las Metodologías Ágiles o “ligeras” constituyen un nuevo enfoque en el desarrollo de software, mejor aceptado por los desarrolladores de proyectos que las metodologías convencionales (ISO-9000, CMM, etc.) debido a la simplicidad de sus reglas y prácticas, su orientación a equipos de desarrollo de pequeño tamaño, su flexibilidad ante los cambios y su ideología de colaboración.

Las metodologías de desarrollo ágiles son adaptativas en lugar de predictivas. Los métodos de ingeniería tienden a intentar planear una parte grande del proceso del software en gran detalle para un plazo largo de tiempo, esto funciona bien hasta que las cosas cambian. Así que su naturaleza es resistirse al cambio. Para estas metodologías, no obstante, el cambio es bienvenido.

Intentan ser procesos que se adaptan y crecen en el cambio, incluso al punto de cambiarse ellos mismos. Son orientadas a la gente y no orientados al proceso. Y su meta es definir un proceso que funcionará bien con cualquiera que lo use.

2.3.1 THE AGILE ALLIANCE

En febrero de 2001, tras una reunión celebrada en Utah (EE.UU.), nace el término ágil aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto. Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas.

Tras esta reunión se creó “The Agile Alliance”, una organización, sin ánimo de lucro, dedicada a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida fue el Manifiesto Ágil, un documento que resume la filosofía ágil, es este manifiesto se valora:

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.
- Desarrollar software que funciona más que conseguir una buena documentación. La regla a seguir es no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante. Estos documentos deben ser cortos y centrarse en lo fundamental.
- La colaboración con el cliente más que la negociación de un contrato. Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
- Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

En la Tabla 2.4 se muestran en forma de tabla las diferencias que existen entre las metodologías ágiles y las tradicionales.

Metodología Ágil	Metodología No Ágil (Tradicional)
Pocos artefactos	Más artefactos
Pocos roles	Más roles
No existe un contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo (además in-situ)	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes
Menos énfasis en la arquitectura	La arquitectura es esencial

Tabla 2.4: Metodologías Ágiles vs. Tradicionales

Dentro de las metodologías ágiles una de las características principales es el uso de historias de usuario. Son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

2.3.2 SCRUM

Por último se va a hacer referencia a la metodología ágil utilizada como base para la realización de este proyecto, Scrum [19] y cómo se ha adaptado a nuestro entorno.

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión, ROI por sus siglas en inglés, para su empresa. Se basa en construir primero la funcionalidad de mayor valor para el cliente siguiendo los principios de inspección continua, adaptación, auto-gestión e innovación.

Con Scrum el cliente se entusiasma y se compromete con el proyecto dado que lo ve crecer iteración a iteración. Asimismo, le permite en cualquier momento realinear el software con los objetivos de negocio ya que puede introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración.

Esta metodología de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades.

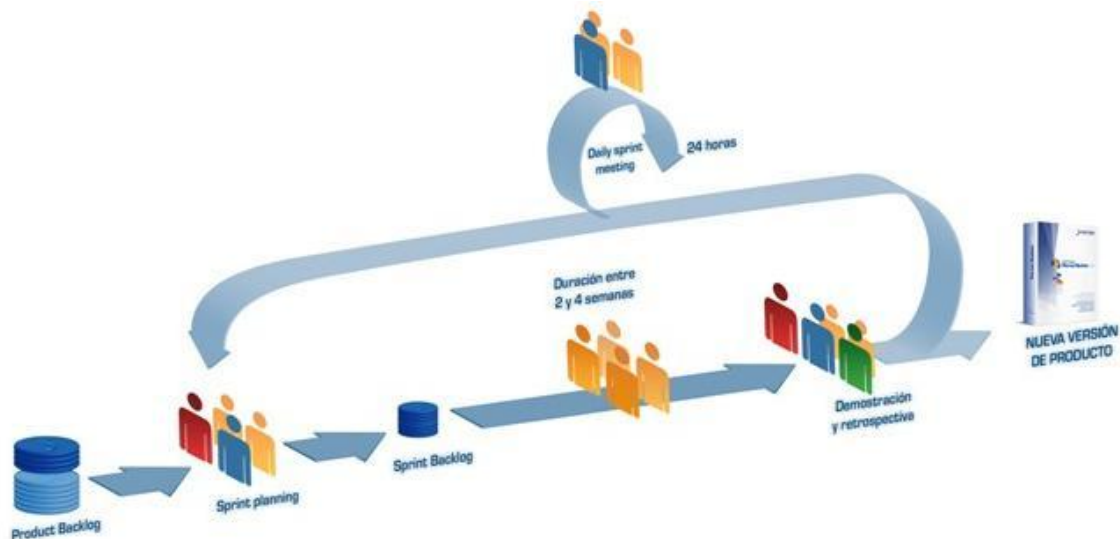


Figura 2.12: Diagrama de flujo de Scrum

Los principales beneficios que proporciona la utilización de Scrum son los siguientes:

- Cumplimiento de expectativas: el cliente establece sus expectativas indicando el valor que le aporta cada requisito/historia del proyecto, el equipo los estima y con esta información establece su prioridad. De manera regular, en las demos de Sprint se comprueba que efectivamente los requisitos se han cumplido y se transmite feedback (retroalimentación) al equipo.
- Flexibilidad a cambios: Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente o evoluciones del mercado.
- Reducción del Time to Market: El cliente puede empezar a utilizar las funcionalidades más importantes del proyecto antes de que esté finalizado por completo.
- Mayor calidad del software: La metódica de trabajo y la necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.
- Mayor productividad: Se consigue entre otras razones, gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.
- Maximiza el ROI: Producción de software únicamente con las prestaciones que aportan mayor valor de negocio gracias a la priorización por retorno de inversión.

- Predicciones de tiempos: Mediante esta metodología se conoce la velocidad media del equipo por sprint (los llamados puntos historia), con lo que consecuentemente, es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad que todavía está como trabajo pendiente.
- Reducción de riesgos: El hecho de llevar a cabo las funcionalidades de más valor en primer lugar y de conocer la velocidad con que el equipo avanza en el proyecto, permite despejar riesgos eficazmente de manera anticipada.

2.4 CONCLUSIONES DEL ANÁLISIS

Analizadas las aplicaciones que se ofrecen servicios de búsqueda y posicionamiento *indoor*, hay una serie de características que se pueden incorporar a la aplicación desarrollada, estas son:

- El uso de un mapa cartográfico como base mostrar los edificios de los campus de la UC3M,
- Permitir búsqueda de edificios y por extensión de lugares ubicados en cada uno de los campus de la UC3M
- Utilización de un plano esquemático dibujado para mostrar la disposición interna de las plantas de los edificios. Por ahora sólo están disponibles los planos arquitectónicos y se puede barajar la posibilidad de dibujarlos con el fin de incorporar los servicios ubicados en cada planta.

Del resto de características señaladas como recomendables en la sección 2.1, como el posicionamiento en interiores y la ruta entre la posición y un servicio o lugar, quedan fuera del alcance del proyecto.

De las características señaladas como deseables es posible incorporar la provisión de servicios de localización a otras apps y la visualización de detalles de lugares.

Existen otras características que no están desarrolladas totalmente en las apps analizadas como la búsqueda de personas ya que la realizan únicamente desde otra app. La finalidad en la aplicación desarrollada es que se pueda buscar personas y lugares en ella y mostrar también la ubicación de las personas e información detallada. Si es cierto que las app analizadas ofrecen servicios de localización usado por aplicaciones de directorio o de eventos pero el servicio de localización es capaz únicamente de ubicar el edificio y no el lugar exacto en el plano del edificio.

En el desarrollo de esta aplicación móvil se va utilizar una metodología de desarrollo ágil, puesto que en un principio, los requisitos de la aplicación no están totalmente definidos. Se elige Scrum por ser la más adecuada a este tipo de desarrollos software, adaptada a iteraciones de dos semanas. Cada iteración se inicia con una reunión de los miembros del equipo de desarrollo. Sobre la última versión del producto, si la hay, se definen las tareas a realizar por cada uno de los miembros del equipo durante la iteración. Al final de la iteración, se obtiene una nueva versión del producto.

Durante el tiempo en el que el equipo de desarrollo lleva a cabo las tareas, se hacen reuniones diarias para comprobar el avance del trabajo, comentar cuestiones técnicas que no hubiesen quedado claras en la reunión de comienzo de la iteración o plantear alternativas a discutir en la siguiente reunión.

CAPÍTULO 3 ANÁLISIS DE LA APLICACIÓN

3.1 INTRODUCCIÓN	36
3.2 DESCRIPCIÓN GENERAL DEL SISTEMA A DESARROLLAR.....	36
3.2.1 PERSPECTIVA DEL PRODUCTO	36
3.2.2 FUNCIONES DEL PRODUCTO	40
3.2.3 CARACTERÍSTICAS DE LOS USUARIOS	40
3.2.4 INTERACCIÓN CON OTROS SISTEMAS	41
3.2.5 RESTRICCIONES	42
3.2.6 SUPOSICIONES Y DEPENDENCIAS	42
3.3 HISTORIAS DE USUARIO	43
3.4 REQUISITOS ESPECÍFICOS	47
3.4.1 REQUISITOS FUNCIONALES	49
3.4.2 INTERFACES EXTERNAS	52
3.4.3 REQUISITOS DE RENDIMIENTO	56
3.4.4 RESTRICCIONES DE DISEÑO	56
3.4.5 ATRIBUTOS DEL SISTEMA.....	57

3.1 INTRODUCCIÓN

En este capítulo se incluye una descripción del sistema que se ha desarrollado junto con las historias de usuario y la especificación de requisitos software. Para su realización se han seguido las partes correspondientes del estándar 830 del IEEE [20].

La descripción general da una visión más amplia que la hasta ahora ofrecida del sistema desarrollado que da solución al problema introducido en el Capítulo 1. Además, sirve como ayuda para dar comienzo a la especificación de las historias de usuario y de los requisitos de software.

Las historias de usuario se definen para conocer cuál es la interacción del usuario con el sistema. No están contempladas en el estándar IEEE 830 pero si en la metodología de desarrollo escogida, SCRUM. Cada historia contiene una serie de tareas que el usuario tiene que realizar para que el sistema le provea una funcionalidad, siendo este el motivo que ha llevado a decidir añadir en las historias de usuario la indicación de los requisitos funcionales y no funcionales con los que cada historia guarda relación.

Finalmente, tras conocer qué es lo que tiene que hacer el sistema y cómo lo tiene que llevar a cabo, se realiza la recogida de requisitos funcionales y no funcionales del sistema.

3.2 DESCRIPCIÓN GENERAL DEL SISTEMA A DESARROLLAR

Tomando el estándar IEEE 830, en esta sección se describen todos los factores que afectan al sistema y a sus requisitos. La finalidad no es describir los requisitos, sino su contexto, permitiendo definirlos con detalle en la sección 3.4 y hacerlos más fácil de entender.

3.2.1 PERSPECTIVA DEL PRODUCTO

El comienzo del planteamiento de un sistema que ofrezca una solución para el problema que se pretende resolver en el presente proyecto, comienza cuando la UC3M decide ofrecer una serie de aplicaciones institucionales para dispositivos móviles orientadas a facilitar la vida universitaria.

Dentro del catálogo de apps que se determina poner a disposición de la comunidad universitaria se encuentran las que se listan a continuación. Todas ellas han sido desarrolladas para los sistemas operativos Android y iOS con la intención de llegar a un público amplio ya que como se analizó en la sección 2.2.1, son los sistemas operativos para dispositivos móviles más utilizados.

- **UC3M Noticias:** ofrece una lista con las últimas noticias de la UC3M. Se puede leer noticias completas y compartirlas en las redes sociales. Permite aumentar el tamaño del texto para que sea accesible a usuarios con dificultades de visión.

- **UC3M Personas:** es un servicio de búsqueda de estudiantes, PDI y PAS sobre el directorio de la UC3M. Ofrece datos relevantes de la persona buscada en el ámbito de la UC3M como por ejemplo, sus datos de contacto. Además, permite añadir como contacto a la persona que se ha buscado.
- **UC3M Multimedia:** es un servicio de visionado de vídeos relacionados con noticias o eventos del ámbito de la UC3M. Clasifica los vídeos en función de su importancia o pertenencia a una temática y los ordena de forma temporal o alfabética. Ofrece un buscador de vídeos.
- **UC3M Estudios:** es un catálogo con la información de las titulaciones de grado y postgrado que se pueden cursar en la UC3M.
- **UC3M Cómo Llegar:** ofrece indicaciones para llegar a los distintos campus de la UC3M en función del medio de transporte que se utilice.
- **UC3M Agenda¹:** es un servicio que muestra los eventos de la UC3M clasificados por su tipo. Ofrece dos tipos de visualización: una vista de lista en la que se ordenan los eventos según su fecha de celebración y otra de calendario que los muestra en el día en que van a tener lugar. Facilita una vista con información relevante de cada evento y permite compartirlos en las redes sociales. Permite aumentar el tamaño del texto para que sea accesible a usuarios con dificultades de visión. Incluye una vista de ayuda para indicar cómo usar la app.
- **UC3M Mapas:** es la app desarrollada como propuesta de solución del presente proyecto. Ofrece una búsqueda geolocalizada de lugares como aulas, despachos, o lugares significativos y de personas. Muestra el campus, edificio y planta del resultado de la búsqueda. Los planos de cada una de las plantas de los edificios de los campus son accesibles, identificando con colores puertas y escaleras. Permite mostrar una vista de detalle de la persona o lugar buscado. Soporta el reporte de errores por email desde la app. Es posible la integración con otras aplicaciones para ofrecer servicios de localización en la UC3M.
- **UC3M Calificaciones:** es un servicio que permite consultar las calificaciones finales que ha obtenido un estudiante durante su vida académica en la UC3M.

Siguiendo la estela de otras universidades españolas y americanas que proveen servicios a la comunidad universitaria utilizando los dispositivos móviles como soporte, se ha desarrollado una app que sirve como contenedor de las apps que se consideran de interés general, distinguiendo así entre aplicaciones autocontenidas y aplicaciones instalables. Las aplicaciones instalables pueden ser descargadas desde el contenedor, app denominada UC3M, y su instalación queda a discreción de los usuarios en función de sus necesidades.

¹ El desarrollo de esta app para el sistema operativo iOS ha sido realizado por el autor del presente proyecto.

Otro criterio de clasificación bajo el que se pueden separar las apps es el público objetivo al que están dirigidos. Las posibilidades son: comunidad universitaria, público externo, estudiantes de la UC3M, estudiantes de nuevo ingreso de la UC3M, personal de la UC3M, nuevo personal de la UC3M y servicios de la UC3M. En la Tabla 3.1 se clasifican las apps desarrolladas por la UC3M bajo los criterios establecidos.

Observando la Tabla 3.1 se aprecia cómo la mayoría de las apps institucionales de la UC3M son autocontenidas y están dirigidas a la comunidad universitaria de la UC3M y a personas externas a la misma. En concreto, UC3M Mapas es una aplicación instalable y está dirigida a usuarios que tienen su primer acercamiento a la Universidad y a aquellos que necesitan hacer uso de servicios de localización en la UC3M en diferentes situaciones de su vida diaria.

Conocido el ecosistema donde se integra el proyecto, es momento de exponer qué valor le aporta a él y a los usuarios que hagan uso de la app implementada. Como ya se ha definido en el capítulo introductorio, el objetivo principal de este proyecto es desarrollar un conjunto de servicios integrados que faciliten la localización indoor de lugares y personas a los usuarios en los diferentes campus de la UC3M. Partiendo de esta base, se puede deducir que el valor principal que aporta a los usuarios es poder conocer la ubicación de cualquier lugar o persona de la UC3M y para el ecosistema, el uso por parte de otras apps de los servicios de localización que provee.

Sobre el valor que aporta a los usuarios, se ve aumentado por dos razones. Por un lado, la aplicación devuelve localizaciones precisas para cada búsqueda y por otro, muestra mapas con sitios de referencia pintados de diferente color al resto del plano de las plantas. La unión de estas dos características resulta en una base para abordar la navegación en interiores como línea de mejora, temática que actualmente suscita gran interés.

Sobre el valor que aporta a otras aplicaciones del ecosistema, un ejemplo de la utilización de los servicios de localización por otra app se puede encontrar en UC3M Agenda. Cuando en UC3M Agenda se está visualizando información sobre un evento y se conoce el lugar de la UC3M donde se va celebrar, el usuario tiene la opción de ver directamente en UC3M Mapas la ubicación del lugar sin necesidad de buscarlo previamente. Esta funcionalidad también es utilizada apps externas al ecosistema como ITGSM12. La funcionalidad principal de esta app es informar de los eventos del VII congreso académico organizado por la asociación ITSMF en la UC3M y para cada evento, la app solicita la ubicación de su lugar de celebración a UC3M Mapas cuando el usuario no sabe dónde está.

Además del valor que UC3M Mapas aporta al ecosistema donde se incorpora y a los usuarios que utilizarán la app, también aporta valor a la UC3M ya que a partir de ahora dispone de una base de datos con la geolocalización de todos los lugares de la UC3M. En un futuro se podrá hacer uso de esta base de datos y de los servicios de búsqueda que se desarrollen con el fin de cubrir nuevas necesidades.

	Distribución		Público objetivo						
	Autocontenida	Instalable	Comunidad universitaria ²	Público externo	Estudiantes UC3M	Nuevos estudiantes UC3M	Personal UC3M	Nuevo personal UC3M	Servicios UC3M
UC3M Noticias	X		X	X					
UC3M Personas	X		X	X					
UC3M Multimedia	X		X	X					
UC3M Estudios	X			X	X				
UC3M Cómo Llegar	X		X			X		X	
UC3M Agenda		X	X	X					
UC3M Mapas		X	X	X		X		X	X
UC3M Calificaciones		X			X				

Tabla 3.1: Clasificación según distribución y público objetivo de las apps UC3M

² En caso de encontrarse una app dirigida a uno o varios estamentos de la UC3M y además a la comunidad universitaria, significa que la app está prioritariamente enfocada hacia los estamentos que se hayan marcado.

3.2.2 FUNCIONES DEL PRODUCTO

Hasta el momento se ha hablado de las necesidades que se han cubierto y de los objetivos que se han marcado en la elaboración de este proyecto. Conocidas ambas partes se pueden definir las funcionalidades principales de la app que se ha desarrollado:

- **Seleccionar preferencias:** los usuarios podrán seleccionar el campus de la UC3M sobre el que desean realizar las búsquedas y el tipo de mapa cartográfico sobre el que se muestran los planos.
- **Buscar lugares:** los usuarios podrán buscar cualquier lugar del campus seleccionado para visualizar la ubicación e información relativa a éste y reportar incidencias de ubicaciones erróneas.
- **Buscar personas:** los usuarios podrán buscar cualquier persona del campus seleccionado para visualizar la ubicación e información relativa a éste y reportar incidencias de ubicaciones erróneas.
- **Ver información:** los usuarios podrán visualizar un manual de ayuda y los créditos de la app e información de contacto.
- **Gestionar peticiones de otras apps:** otras apps podrán solicitar el servicio de búsqueda de un lugar o persona siendo en UC3M Mapas donde se muestre el resultado de la petición.

Se ha trabajado sobre las funciones del producto para obtener los requisitos funcionales y no funcionales de partida del sistema. Tal como se dijo en la sección 1.4 en la que se aborda el método de resolución, los requisitos se han definido a medida que han surgido y discutido diferentes opciones.

3.2.3 CARACTERÍSTICAS DE LOS USUARIOS

La definición del perfil de usuario que va a utilizar la app desarrollada corresponde al público objetivo presentado en el apartado 3.2.1. Como requisito previo, todos los usuarios que deseen utilizar la app deberán disponer de un dispositivo móvil con el sistema operativo iOS.

El primer grupo de público objetivo es el compuesto por las personas que comienzan una relación académica o laboral con la UC3M. Estos usuarios no conocen dónde están ubicados los lugares de los campus de la UC3M ni la distribución interior de los edificios. Se diferencian entre estudiantes de nuevo ingreso y nuevo personal de PDI y de PAS.

El segundo grupo de público objetivo es el compuesto por las personas que actualmente mantienen una relación académica o laboral con la UC3M. Estos usuarios conocen el campus en el que normalmente desempeñan su actividad pero pueden no conocer algunas partes de él. Cuando estos usuarios se encuentran en un campus diferente al suyo, pueden no conocerlo y necesitar los servicios de localización. Se diferencian entre estudiantes y personal de PDI y de PAS.

El tercer grupo de público objetivo es el compuesto por los servicios y unidades de la UC3M que quieren incluir su ubicación en la base de datos externa que va a

utilizar la app. A priori, los identificadores y ubicaciones de todos los lugares de la UC3M están incluidos en la base de datos pero es posible que se quiera incluir también el nombre asociado al servicio o unidad. Este grupo también está abierto a otros colectivos con identidad propia en la UC3M como asociaciones o institutos.

3.2.4 INTERACCIÓN CON OTROS SISTEMAS

La app interactúa con varios sistemas que se pueden englobar en tres grupos: de soporte y provisión de servicios de localización, de comunicación interna y de comunicación externa.

Dentro del grupo de soporte y provisión de servicios de localización, se encuentran sistemas ya desarrollados que facilitan mostrar los mapas de las plantas de los edificios y recuperar localizaciones. Estos sistemas son:

- Servicio de mapas cartográficos: se necesita el acceso a un servicio de mapas cartográficos para que el usuario pueda visualizar los campus de la UC3M y para que la aplicación lo utilice como base a la hora de mostrar los planos de las plantas de los edificios. El plano de la planta de un edificio se sitúa sobre la captura del edificio que provee el servicio de mapas cartográficos.

En iOS el acceso a sistemas de cartografía se puede hacer de forma nativa desde el framework MapKit. El framework se encarga de realizar las peticiones al sistema de cartografía que hasta la versión 6 de iOS, ha sido Google Maps. A partir de la versión 6 el sistema al que se realizan las peticiones es Maps, desarrollado por Apple.

- Directorio UC3M: se hacen peticiones al servicio web que provee esta aplicación web de la UC3M para obtener información sobre una persona en la que se incluye, entre otros, el lugar o lugares que tiene asignados en la UC3M.
- SEL UC3M Mapas: se hacen peticiones al servicio web que provee esta aplicación web desarrollada por el grupo de investigación SEL-UC3M para recuperar la localización geográfica de los lugares de la UC3M.

Hay que tener especial cuidado cuando se utilizan servicios externos porque no podemos controlar su disponibilidad ya que parte o la totalidad de las funcionalidades de la app podrían dejar de funcionar. En este caso, o bien el acceso al mantenimiento del servicio no es posible como sucede con el servicio de cartografía, aunque si es un servicio fiable, o el acceso se puede realizar mediante una persona que guarde relación con el ámbito del proyecto como sucede con los servicios de la UC3M y del SEL-UC3M.

Dentro del grupo de comunicación interna se encuentran las comunicaciones que realiza la app con otras apps instaladas en el dispositivo. Estas comunicaciones se realizan a través de diferentes *frameworks* permitiendo llamar por teléfono a una persona, añadirla como contacto o mandarla un correo, y compartir la app en las redes sociales.

Dentro del grupo de comunicación externa se encuentran las comunicaciones que recibe la app. Se ha provisto a la app de un protocolo genérico para que sea capaz de procesar peticiones que le llegan desde otras apps. El resto de apps únicamente tienen que conocer los parámetros que tienen que facilitar a UC3M Mapas. Las apps UC3M Agenda e ITGSM12 usan este protocolo para solicitar la localización del lugar de celebración de los eventos que acontecen en la UC3M.

3.2.5 RESTRICCIONES

Para fijar el alcance del sistema y hacer constar de manera implícita aspectos del mismo que no tienen cabida en otros apartados se definieron las siguientes restricciones:

La interfaz gráfica de la aplicación deberá cumplir con las guías de diseño [22] y de desarrollo [23] de Apple.

La aplicación deberá ser compatible con dispositivos cuya versión de iOS sea igual o superior a la 4.2.

El lenguaje de programación que se tiene que usar para programar la app es Objective-C.

El sistema UC3M Mapas deberá disponer de un mecanismo a través del cual permitir a otras apps solicitar servicios de localización. Se definirán los parámetros necesarios para proveer el servicio.

Las comunicaciones con los servidores de la UC3M y del SEL-UC3M se realizarán a través de un servicio web que devuelve los datos solicitados en formato JSON. La app deberá procesar estos datos para mostrar los resultados de las consultas.

Los planos de las plantas de los edificios que se guarden en el repositorio deberán cumplir los requisitos establecidos en el OpenGIS Web Map Tile Service Implementation Standard [24]. Es un estándar de facto que se encarga de definir cómo almacenar la estructura de directorios de almacenamiento de tiles.

La criticidad del sistema reside en el tiempo de respuesta de las consultas.

3.2.6 SUPOSICIONES Y DEPENDENCIAS

Como se ha descrito en el apartado 3.2.4, la app interacciona con otros sistemas con el fin de obtener soporte y provisión para los servicios de localización. Esta interacción crea una dependencia fuerte en el sistema ya que parte de su funcionalidad principal se pierde si dejan de estar disponibles.

Las principales dependencias de la app son:

- **Servicio web SEL-UC3M:** en caso de que el servicio web provisto por el grupo SEL no estuviera disponible, la aplicación no podría realizar ningún tipo de búsqueda ya que la conecta con la base de datos de lugares georreferenciados y con el directorio de personas de la UC3M

- **Servicio de directorio UC3M:** en caso de que el servicio de directorio provisto por la UC3M no estuviera disponible, la aplicación no podría ofrecer la búsqueda personas porque el servicio web del grupo SEL no tendría acceso ni a los datos del lugar en la UC3M ni a los datos de contacto de la persona buscada.
- **Servicio de cartografía:** en caso de que el servicio de cartografía del que hace uso iOS no estuviera disponible, la aplicación no podría ofrecer ningún tipo de búsqueda porque no sería posible mostrar un plano superpuesto sobre el edificio correspondiente para mostrar la ubicación de la persona o del lugar buscado.

3.3 HISTORIAS DE USUARIO

En esta sección se utilizan las historias de usuario para detallar con un lenguaje cercano a cualquier persona ajena al desarrollo de sistemas, las funcionalidades que la aplicación desarrollada debe proveer. Su uso viene motivado porque en la metodología SCRUM, se utilizan para describir las características que el usuario espera que tenga el sistema.

Las historias de usuario son las secuencias de interacción del usuario con el sistema. Se descomponen en tareas que deberán de ser realizadas por el usuario para acceder y/o lograr la funcionalidad de la que desea hacer uso. Las historias de usuario deben de caracterizarse por ser independientes, negociables, valorables, estimables, pequeñas y comprobables.

Las historias de usuario pueden incorporar tanto aspectos vinculados con la funcionalidad del sistema como cualquier otro aspecto del mismo ya que están realizadas desde el punto de vista del usuario.

El formato de tabla que se utiliza en el presente documento para describir las historias de usuario es el siguiente:

ID	ESFUERZO
OBJETIVO	
SECUENCIA DE PASOS	
REQUISITOS FUNCIONALES	
REQUISITOS NO FUNCIONALES	

Tabla 3.2: Formato de tabla de historia de usuario

Cada uno de los campos tendrá por contenido:

- **ID:** el identificador de la historia de usuario. Siguen el formato HU-XXX.
- **ESFUERZO:** dedicación, en horas, invertidas para implementar la funcionalidad que satisface.
- **OBJETIVO:** funcionalidad del sistema que satisface.
- **SECUENCIA DE PASOS:** pasos que tiene que dar el usuario para lograr la funcionalidad.
- **REQUISITOS FUNCIONALES:** requisitos funcionales extraídos de la historia de usuario.
- **REQUISITOS NO FUNCIONALES:** requisitos no funcionales extraídos de la historia de usuario.

A continuación se las historias de usuario del sistema que se han definido:

ID	HU-001	ESFUERZO	15 horas
OBJETIVO	Seleccionar las opciones de configuración de visualización de mapa y el campus sobre el que realizar las búsquedas.		
SECUENCIA DE PASOS	<ol style="list-style-type: none"> 1. El usuario inicia la app. 2. El usuario pulsa el botón de configuración. 3. El usuario elije el tipo de mapa que desea visualizar. 4. El usuario elije el campus sobre el que realizar las búsquedas. 5. El usuario pulsa el botón de finalización. 		
REQUISITOS FUNCIONALES	RF-010 - Seleccionar campus RF-011 - Seleccionar tipo de mapa		
REQUISITOS NO FUNCIONALES	RI-004 - Vista de configuración		

Tabla 3.3: HU-001 - Seleccionar opciones de configuración

ID	HU-002	ESFUERZO	285 horas
OBJETIVO	Buscar un lugar de un campus de la UC3M.		
SECUENCIA DE PASOS	<ol style="list-style-type: none"> 1. El usuario inicia la app. 2. El usuario, si lo desea, centra su posición en el mapa. 3. El usuario selecciona la búsqueda de tipo Lugar. 4. El usuario introduce en la barra de búsqueda los datos de los que dispone. 5. El usuario pulsa el botón de búsqueda. 6. El usuario elige, en caso de que se ofrezcan varios resultados para los datos introducidos, el lugar que le interesa. 7. El usuario pulsa el globo de texto del lugar si quiere ver los detalles del lugar. 8. El usuario elige una de las acciones disponibles en la vista de detalle del lugar. 		
REQUISITOS FUNCIONALES	RF-001 - Seleccionar el tipo de búsqueda RF-002 - Buscar según un término de búsqueda RF-003 - Mostrar detalles de un lugar RF-005 - Reportar incidencia RF-012 - Visualizar posición actual		
REQUISITOS NO FUNCIONALES	RI-003 - Vista principal RI-005 - Vista de detalle de lugar RI-010 - Comunicación con el servicio web del SEL-UC3M RR-001 - Número máximo de resultados de búsqueda		

Tabla 3.4: HU-002 - Buscar un lugar

ID	HU-003	ESFUERZO	285 horas
OBJETIVO	Buscar una persona de un campus de la UC3M.		
SECUENCIA DE PASOS	<ol style="list-style-type: none"> 1. El usuario inicia la app. 2. El usuario, si lo desea, centra su posición en el mapa. 3. El usuario selecciona la búsqueda de tipo Persona. 4. El usuario introduce en la barra de búsqueda los datos de los que dispone. 5. El usuario pulsa el botón de búsqueda. 6. El usuario elige, en caso de que se ofrezcan varios resultados para los datos introducidos, la persona que le interesa. 7. El usuario pulsa el globo de texto de la persona si quiere ver los detalles de la persona. 8. El usuario elige una de las acciones disponibles en la vista de detalle de la persona. 		
REQUISITOS FUNCIONALES	RF-001 - Seleccionar el tipo de búsqueda RF-002 - Buscar según un término de búsqueda RF-004 - Mostrar detalles de una persona RF-005 - Reportar incidencia RF-006 - Enviar correo electrónico a una persona RF-007 - Llamar a una persona RF-008 - Agregar a una persona como contacto RF-012 - Visualizar posición actual		
REQUISITOS NO FUNCIONALES	RI-003 - Vista principal RI-006 - Vista de detalle de persona RI-010 - Comunicación con el servicio web del SEL-UC3M RR-001 - Número máximo de resultados de búsqueda		

Tabla 3.5: HU-003 - Buscar una persona

ID	HU-004	ESFUERZO	7,5 horas
OBJETIVO	Visualizar la ayuda y los créditos		
SECUENCIA DE PASOS	<ol style="list-style-type: none">1. El usuario inicia la app.2. El usuario pulsa el botón de información.3. El usuario elige uno de los contenidos de información.		
REQUISITOS FUNCIONALES	RF-009 - Mostrar créditos y ayuda		
REQUISITOS NO FUNCIONALES	RI-007 - Vista de acerca de RI-008 - Vista de créditos RI-009 - Vista de ayuda		

Tabla 3.6 HU-004 - Visualizar ayuda y créditos

ID	HU-005	ESFUERZO	7,5 horas
OBJETIVO	Proveer la localización de un lugar solicitada por otra app.		
SECUENCIA DE PASOS	<ol style="list-style-type: none">1. El usuario selecciona la opción de visualización en mapa desde otra app.		
REQUISITOS FUNCIONALES	RF-013 - Proveer servicio de localización		
REQUISITOS NO FUNCIONALES	RI-001- API de comunicación		

Tabla 3.7: HU-005 - Proveer localización a app externa

3.4 REQUISITOS ESPECÍFICOS

Los requisitos que se definen en esta sección poseen el nivel de detalle necesario para que los diseñadores puedan diseñar un sistema que satisfaga estos requisitos y que permita al equipo de pruebas, planificar y realizar las pruebas que demuestren si el sistema cumple o no los requisitos. Los requisitos definidos describen comportamientos externos del sistema perceptibles por parte de los usuarios, operadores y otros sistemas.

En la sección 3.4.1 se incluye la definición los requisitos funcionales que son los que tratan sobre la funcionalidad que proporciona la aplicación. En otras palabras, definen qué hace la aplicación.

En la sección 3.4.2 se incluye la definición de requisitos de interfaces externas. Por interfaces externas se entienden tanto las vistas de la aplicación con las que puede interactuar directamente el usuario, como mecanismos que se ofrecen a otras aplicaciones para conectarse con UC3M Mapas y las utilidades que conectan la aplicación con servicios externos.

En la sección 3.4.3 se incluye la definición de los requisitos que afectan al rendimiento de la aplicación.

En la sección 3.4.4 se incluyen las restricciones de diseño que limitan ciertos puntos en la aplicación.

Para definir los requisitos se utiliza una tabla con una serie de campos cuyos posibles valores se muestran en la Tabla 3.8.

ID		Nombre	
Dependencias			
Descripción			

Tabla 3.8. Formato de tabla de especificación de requisito

Los campos y el contenido de los mismos de la tabla utilizada para definir los requisitos son:

- **ID:** el identificador de la historia de usuario. Siguen el formato COD-XXX. Para los requisitos funcionales COD es RF, para los de interfaz externa es RI, para los de rendimiento RR y para los de restricción de diseño es RD.
- **Nombre:** nombre del requisito. Es deseable que el nombre resuma el requisito y que incluya quién hace la acción, la acción y los complementos de la acción.
- **Dependencias:** identificador de los requisitos de los que depende para poder ser implementado.
- **Descripción:** explicación del requisito que permita a cualquier programador implementarlo sin necesidad de consultar más información.

Aunque para la definición de requisitos se ha seguido el estándar IEE 830, se ha decidido no incluir algunos campos que se incluyen tradicionalmente en la recogida de requisitos como su prioridad o necesidad debido al tipo de metodología utilizada. Como SCRUM es una metodología ágil, no es posible plasmar la dinámica de evolución de los valores de estos campos a lo largo de las iteraciones. Además, una vez que se ha desarrollado la aplicación no resulta necesario conocer su valor.

3.4.1 REQUISITOS FUNCIONALES

ID	RF-001	Nombre	El usuario selecciona el tipo de búsqueda.
Dependencias			
Descripción	La aplicación deberá posibilitar la selección del tipo de búsqueda, lugar o persona, de manera que dependiendo de la opción seleccionada, se realice un tipo u otro de búsqueda.		

Tabla 3.9: RF-001 - Selección del tipo de búsqueda

ID	RF-002	Nombre	La aplicación busca un término de búsqueda indicado por el usuario.
Dependencias	RF-001, RF-010, RF-011		
Descripción	La aplicación deberá mostrar en el mapa la ubicación correspondiente al término de búsqueda indicado, teniendo en cuenta el tipo de búsqueda, el campus y el tipo de plano cartográfico seleccionados. En caso de búsqueda de persona: <ul style="list-style-type: none">• Deberá mostrar la ubicación del despacho que tiene asignado.• Si no tiene despacho asignado, deberá indicar que la persona buscada no se ha encontrado en ese campus. En caso de búsqueda de lugar: <ul style="list-style-type: none">• Deberá mostrar la ubicación del lugar.• Si no se encuentra, deberá indicar que el término de búsqueda no se ha encontrado en ese campus.		

Tabla 3.10: RF-002 - Buscar según un término de búsqueda

ID	RF-003	Nombre	El usuario visualiza los detalles de un lugar buscado.
Dependencias	RF-002		
Descripción	La aplicación deberá permitir consultar los detalles del lugar que se ha obtenido como resultado tras una búsqueda de lugar. Los datos que la aplicación deberá mostrar son: despacho, planta, edificio y campus.		

Tabla 3.11: RF-003 - Mostrar detalles de un lugar

ID	RF-004	Nombre	El usuario visualiza los detalles de una persona buscada.
Dependencias	RF-002		
Descripción	La aplicación deberá permitir consultar los detalles de la persona que se ha obtenido como resultado tras una búsqueda de persona. Los datos que la aplicación deberá mostrar son: nombre y apellidos, departamento/unidad, correo electrónico, despacho, edificio, campus y teléfono.		

Tabla 3.12: RF-004 - Mostrar detalles de una persona

ID	RF-005	Nombre	El usuario reporta una incidencia al administrador de la app.
Dependencias	RF-003, RF-004		
Descripción	La aplicación deberá permitir reportar una incidencia al administrador de la aplicación mediante la app predeterminada de correo electrónico de iOS.		

Tabla 3.13: RF-005 - Reportar incidencia

ID	RF-006	Nombre	El usuario envía un correo electrónico a una persona buscada.
Descripción	La aplicación deberá permitir abrir la app predeterminada de correo de iOS para enviar un mensaje a una persona buscada.		

Tabla 3.14: RF-006 - Enviar correo electrónico a una persona

ID	RF-007	Nombre	El usuario llama a una persona buscada.
Dependencias	RF-004		
Descripción	La aplicación deberá permitir abrir la app de teléfono de iOS para llamar a una persona buscada.		

Tabla 3.15: RF-007 - Llamar a una persona

ID	RF-008	Nombre	El usuario agrega a una persona buscada como contacto del terminal.
Dependencias	RF-004		
Descripción	La aplicación deberá permitir agregar a una persona como contacto en la agenda de contactos del terminal.		

Tabla 3.16: RF-008 - Agregar a una persona como contacto

ID	RF-009	Nombre	El usuario visualiza los créditos y la ayuda de la app.
Dependencias			
Descripción	La aplicación deberá permitir abrir los créditos y la ayuda de la app. Los créditos deberán incluir un e-mail de contacto, los nombres de los implicados en la realización del proyecto y el nombre de usuario del grupo SEL en Twitter.		

Tabla 3.17: RF-009 - Mostrar créditos y ayuda

ID	RF-010	Nombre	El usuario selecciona el campus sobre el que realizar las búsquedas.
Dependencias			
Descripción	La aplicación deberá permitir seleccionar el campus en el que se desean realizar las búsquedas. Los campus permitidos deberán ser Leganés, Getafe y Colmenarejo.		

Tabla 3.18: RF-010 - Seleccionar campus

ID	RF-011	Nombre	El usuario selecciona el tipo de mapa cartográfico.
Dependencias			
Descripción	La aplicación deberá permitir seleccionar el tipo de mapa cartográfico que se desea visualizar. Los tipos permitidos deberán ser Normal, Satélite e Híbrido.		

Tabla 3.19: RF-011 - Seleccionar tipo de mapa

ID	RF-012	Nombre	El usuario visualiza su posición actual.
Dependencias			
Descripción	La aplicación deberá permitir conocer la posición GPS actual del usuario.		

Tabla 3.20: RF-012 - Visualizar posición actual

ID	RF-013	Nombre	La aplicación provee servicio de localización en la UC3M a otra app.
Dependencias			
Descripción	La aplicación deberá permitir a otras apps solicitar la localización de lugares de la UC3M.		

Tabla 3.21: RF-013 - Proveer servicio de localización

3.4.2 INTERFACES EXTERNAS

Cabe recordar que según el estándar IEE 830, se definen como requisitos de interfaz los relacionados con la provisión por parte de la aplicación de servicios a otras aplicaciones móviles, con las vistas de la aplicación con las que puede interactuar directamente el usuario y con las utilidades para conectar la aplicación con servicios externos.

ID	RI-001	Nombre	La aplicación provee una API de comunicación para otras apps.
Dependencias	RF-013		
Descripción	La aplicación deberá ser capaz de recibir parámetros de otras apps iOS mediante un protocolo conocido como URL-Schema. Los parámetros deberán indicar la función de búsqueda a utilizar, de lugar o de persona, el campus sobre el que se va a realizar la búsqueda y el término de búsqueda a buscar.		

Tabla 3.22: RI-001 - API de comunicación

ID	RI-002	Nombre	La aplicación ofrece las vistas de usuario en castellano y en inglés.
Dependencias			
Descripción	Las interfaces de usuario de la aplicación deberán de tener soporte para los idiomas castellano e inglés. El idioma de los textos variará en función de las opciones de configuración del terminal del usuario. Quedan excluidos los datos que se reciban de los servicios que utilice la aplicación.		

Tabla 3.23: RI-002 - Idioma de la aplicación

ID	RI-003	Nombre	La aplicación ofrece una vista principal de búsqueda.
Dependencias	RF-001, RF-002, RF-012, RI-002, RI-010		
Descripción	La aplicación deberá ofrecer una vista principal con un botón de configuración, un botón de información, un campo de entrada de texto para las búsquedas y un mapa cartográfico. Sobre el mapa deberá superponer un texto en el que se indique el campus en el que se está buscando y, en caso de que se muestre algún plano, el edificio y la planta a la que pertenece y sobre el plano, la ubicación del lugar o la persona buscado junto con un globo informativo. Además deberá mostrar controles para la selección entre búsqueda de lugar o de persona y conocer la posición actual.		

Tabla 3.24: RI-003 - Vista principal

ID	RI-004	Nombre	La aplicación ofrece una vista de configuración.
Dependencias	RF-010, RF-011, RI-002, RI-003		
Descripción	La aplicación deberá ofrecer una vista que permita seleccionar el tipo de plano a mostrar, normal, satélite o híbrido y el campus en el que se desea buscar, Leganés, Getafe o Colmenarejo.		

Tabla 3.25: RI-004 - Vista de configuración

ID	RI-005	Nombre	La aplicación ofrece una vista de detalle de lugar.
Dependencias	RF-003, RF-005 RI-002, RI-003		
Descripción	<p>La aplicación deberá ofrecer una vista que incluya el identificador, el despacho, la planta, el edificio y el campus de un lugar buscado.</p> <p>La aplicación también deberá permitir reportar una incidencia del lugar dando acceso a la app de correo. El correo deberá predefinirse con un destinatario, un asunto y un esquema de texto que pretenda servir como guía para reportar el error correctamente.</p>		

Tabla 3.26: RI-005 - Vista de detalle de lugar

ID	RI-006	Nombre	La aplicación ofrece una vista de detalle de persona.
Dependencias	RF-004, RF-005, RF-006, RF-007, RF-008, RI-002, RI-003		
Descripción	<p>La aplicación deberá ofrecer una vista que incluya el nombre y apellidos, departamento/unidad, correo electrónico, despacho, edificio, campus y teléfono de una persona buscada. Deberá dar acceso a mandar un correo a la persona a la dirección de correo y llamarla por teléfono al teléfono que muestra y agregarla como contacto en el dispositivo.</p> <p>La aplicación también deberá permitir reportar una incidencia del lugar dando acceso a la app de correo. El correo deberá predefinirse con un destinatario, un asunto y un esquema de texto que pretenda servir como guía para reportar el error correctamente.</p>		

Tabla 3.27: RI-006 - Vista de detalle de persona

ID	RI-007	Nombre	La aplicación muestra una vista de créditos y ayuda.
Dependencias	RF-009, RI-002, RI-003		
Descripción	La aplicación deberá mostrar una vista que facilite los siguientes datos acerca de la aplicación: versión, dirección de correo electrónico de contacto, usuario de Twitter del grupo de investigación que la ha desarrollado, @seluc3m. También deberá permitir la visualización de todas las partes involucradas en el desarrollo y el acceso a la ayuda de la app.		

Tabla 3.28: RI-007 - Vista de acerca de

ID	RI-008	Nombre	La aplicación muestra una vista con las partes involucradas en el desarrollo del proyecto.
Dependencias	RI-007		
Descripción	La aplicación deberá mostrar una vista en la que se incluya el logotipo de la universidad, una dirección de correo electrónico corporativa y el nombre de los vicerrectorados, grupos de investigación y servicios de la UC3M involucrados en la creación de la aplicación.		

Tabla 3.29: RI-008 - Vista de créditos

ID	RI-009	Nombre	La aplicación muestra una vista con información de utilización de la misma.
Dependencias	RI-007	Fecha	
Descripción	La aplicación deberá mostrar una vista que muestre indicaciones de cómo llevar a cabo sus funcionalidades básicas.		

Tabla 3.30: RI-009 - Vista de ayuda

ID	RI-010	Nombre	La aplicación comunica con el servicio web del SEL-UC3M.
Dependencias	RF-002		
Descripción	La aplicación se deberá conectar al servicio web del SEL-UC3M a través de una petición GET para realizar las consultas de búsqueda de lugares y personas con los siguientes parámetros: <ul style="list-style-type: none">• <i>function</i> (find person2)• <i>idCampus</i> (1 → Leganés 2 → Getafe 3 → Colmenarejo)• <i>term</i> (el término que desee buscar el usuario)		

Tabla 3.31: RI-010 – Comunicación con el servicio web del SEL-UC3M

3.4.3 REQUISITOS DE RENDIMIENTO

ID	RR-001	Nombre	La aplicación muestra un número limitado de resultados de búsqueda.
Dependencias	RF-002, RI-003, RI-010		
Descripción	La aplicación deberá mostrar como máximo los diez primeros resultados de lugares cuando se trate de una búsqueda de lugar o los quince primeros resultados de personas cuando se trate de una búsqueda de persona.		

Tabla 3.32: RR-001 – Número máximo de resultados de búsqueda

3.4.4 RESTRICCIONES DE DISEÑO

ID	RD-001	Nombre	El programador implementa la aplicación compatible a partir de una versión del sistema operativo iOS.
Dependencias			
Descripción	La aplicación deberá ser compatible con los dispositivos móviles Apple que usen la versión 4.2 de iOS en adelante.		

Tabla 3.33: RD-001 - Versión del Sistema Operativo

3.4.5 ATRIBUTOS DEL SISTEMA

La aplicación deberá ser fiable y para ello deberá tener en cuenta todos los posibles errores controlables desde ella misma y mostrar al usuario información relativa a estos errores de manera que el usuario pueda intentar solucionarlos si dependen de él.

La aplicación deberá poder ser mantenida sin mucha dificultad por parte de otro desarrollador utilizando la documentación disponible.

La aplicación no deberá ser portable ya que en este caso es específica para el sistema operativo iOS. Por el contrario, sí deberá ser compatible con el mayor número de versiones de este sistema operativo.

La aplicación deberá estar preparada para responder frente a ataques. En este caso, la capa de seguridad está integrada en la parte del servidor, de manera que en la aplicación móvil no es necesario implementar aspectos de seguridad.

CAPÍTULO 4 DISEÑO DE LA APLICACIÓN

4.1	PAPER PROTOTYPE	60
4.1.1	PRIMERA VERSIÓN DEL PAPER PROTOTYPE.....	61
4.1.2	ÚLTIMA VERSIÓN DEL PAPER PROTOTYPE.....	62
4.2	ALTERNATIVAS DE DISEÑO	69
4.2.1	ELEMENTOS DEL SISTEMA.....	69
4.2.2	PRIMERA ALTERNATIVA DE DISEÑO.....	72
4.2.3	SEGUNDA ALTERNATIVA DE DISEÑO	73
4.2.4	TERCERA ALTERNATIVA DE DISEÑO	74
4.3	DIAGRAMAS DE COMPONENTES Y DE CLASES	75
4.4	DIAGRAMAS DE SECUENCIA	81
4.5	DISEÑO DE LA BASE DE DATOS.....	87
4.6	DISEÑO DEL REPOSITORIO DE PLANOS	89

En este capítulo se presentan en primer lugar, los *paper prototypes*, o prototipos en papel en su traducción al español, modificados progresivamente en cada una de las iteraciones en las que se establecieran cambios que afectaran de manera considerable a la interfaz de usuario. Los *paper prototypes* son muy útiles para expresar gráficamente los requisitos definidos para una aplicación, permitiendo enseñar al cliente el aspecto de la aplicación y ofrecerle el flujo de interacción del usuario con la misma. Las demostraciones que se realicen posibilitarán discutir, mejorar e incluir requisitos antes de que se pase a una siguiente etapa de diseño o de implementación.

Después de presentar los *paper prototypes*, se describen las alternativas de diseño del sistema que se han barajado, incluyendo sus ventajas e inconvenientes e indicando cuál ha sido la alternativa elegida y explicando el por qué de su elección.

Conocido el diseño del sistema, se muestra el diagrama de componentes para ofrecer una vista panorámica del diseño realizado y en la siguiente sección, se detalla el diagrama de clases de la aplicación en el cual se observan las relaciones entre las clases involucradas en el diagrama de componentes.

Una vez mostrados los diagramas de clases, se presentan los diagramas de secuencia necesarios para que un programador sea capaz de implementar la interacción de los objetos del sistema.

También se explica brevemente, y mediante un diagrama Entidad/Relación y un diagrama relacional, el diseño de la base de datos de lugares georreferenciados con la que interactuará la aplicación a través del servicio web del SEL-UC3M.

Por último, se comenta el diseño elegido para la estructura que deberá tener el repositorio de planos y el proceso de edición de los planos originales de los edificios de la UC3M.

4.1 PAPER PROTOTYPE

En esta sección se muestra el *paper prototype* de la aplicación. Los *paper prototypes* son un boceto de las interfaces de usuario de una aplicación y pueden ser realizados a mano o con un software de edición gráfica. Aportan dos ventajas: son fáciles y poco costosos de realizar y ofrecen al cliente una visión general del aspecto y de la interacción con la aplicación sin necesidad de añadir funcionalidad. También son de ayuda a diseñadores y programadores para poder transmitirse su visión de la aplicación.

Durante el desarrollo de la aplicación, el *paper prototype* inicial realizado para UC3M Mapas ha evolucionado. La primera versión se realizó al inicio del proyecto y en las diferentes iteraciones se añadieron y eliminaron elementos de la interfaz de usuario a favor de mejorar la experiencia de usuario.

4.1.1 PRIMERA VERSIÓN DEL PAPER PROTOTYPE

En la primera iteración de desarrollo se realizaron los prototipos comprendidos entre la Figura 4.1 y la Figura 4.4. En la Figura 4.1 se ve la vista principal de la app compuesta por una barra de búsqueda para introducir los términos de búsqueda y ejecutar la búsqueda, un botón de configuración y un mapa cartográfico sobre el que superponer los planos de las plantas y la ubicación del resultado de la búsqueda.



Figura 4.1: Vista principal (versión 1)

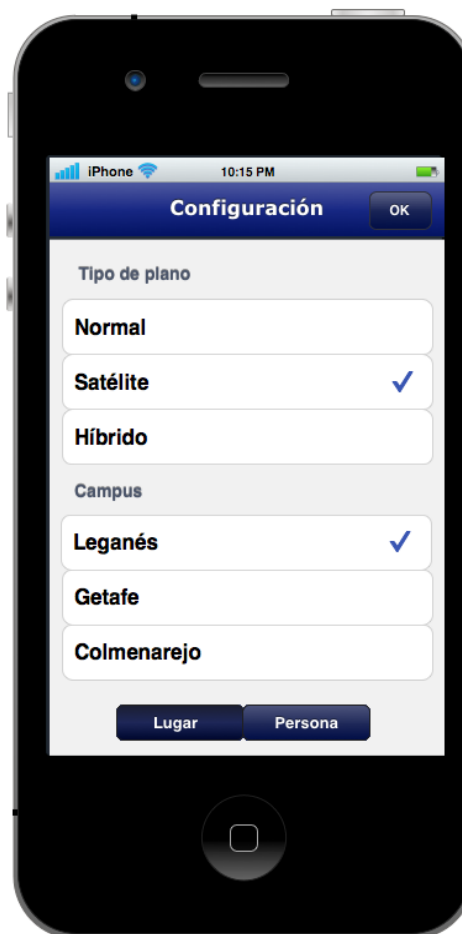


Figura 4.2: Configuración (versión 1)

El usuario puede cambiar las opciones de configuración para elegir el tipo de plano cartográfico sobre el que mostrar los planos, el campus donde realizar las búsquedas y el tipo de búsqueda a realizar. La interfaz se puede observar en la Figura 4.2 y se ve como para elegir el tipo de plano y el campus, se propone el uso de tablas que permitan selección de filas y para indicar el tipo de búsqueda, un botón segmentado en el que el botón más oscuro marca la elección del usuario.

Cuando el usuario introduce un término de búsqueda en la barra de búsqueda y la ejecuta pulsando la lupa, la app muestra sobre el mapa cartográfico el plano de la planta del edificio en el que se encuentra el resultado tal como se ve en la Figura 4.3. Se

propone utilizar los planos esquemáticos que actualmente se encuentran físicamente en los descansillos de cada planta de los edificios de la UC3M como planos a superponer en el mapa cartográfico.

El usuario puede colocar dos dedos sobre la pantalla del dispositivo y juntarlos o alejarlos deslizándolos sobre la misma para aumentar o disminuir el tamaño del mapa y los planos. En la Figura 4.4 se muestra un ejemplo si el usuario decide aumentar el tamaño.



Figura 4.3: Resultado de búsqueda de un lugar (versión 1)

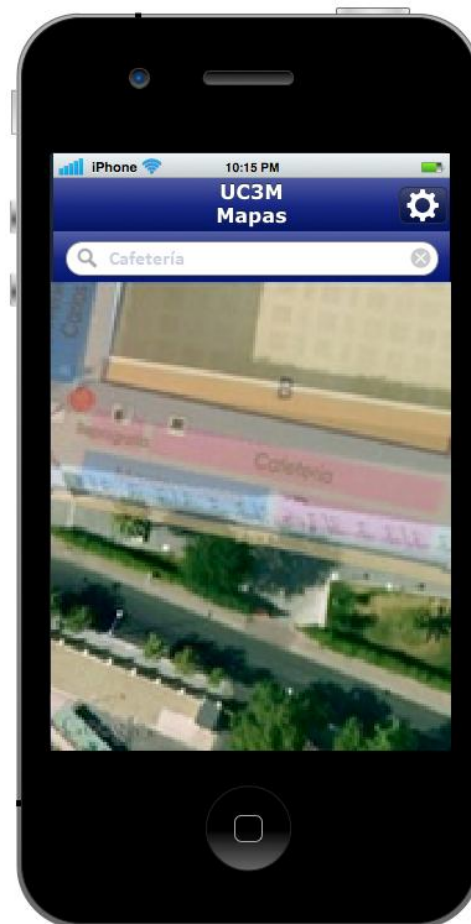


Figura 4.4: Ampliación de resultado de búsqueda de un lugar (versión 1)

4.1.2 ÚLTIMA VERSIÓN DEL PAPER PROTOTYPE

La última versión del *paper prototype* de la aplicación se realizó en junio de 2012 y con respecto a la primera versión, se pueden observar modificaciones significativas en el diseño, en parte propiciadas por la aparición de nuevos requisitos. El *paper prototype* final es el que comprende desde la Figura 4.5 a la Figura 4.16 y se puede verificar que es bastante similar al aspecto final de la app.

En la Figura 4.5 se pueden advertir los cambios en la vista principal con respecto a la primera versión. En la barra superior aparece además del botón de configuración, otro para acceder a la información de la app. Se mantiene la barra de búsqueda, la cual incorpora un ejemplo de término de búsqueda, y la vista del mapa cartográfico al que como novedad, se superpone un dibujo del campus para que sus límites sean identificados por los usuarios. Debajo de la barra de búsqueda y superpuesto al contenedor del mapa, se indica en qué campus se va a realizar una búsqueda y, en caso de haber encontrado un lugar o persona, el edificio y la planta del plano que se muestre.

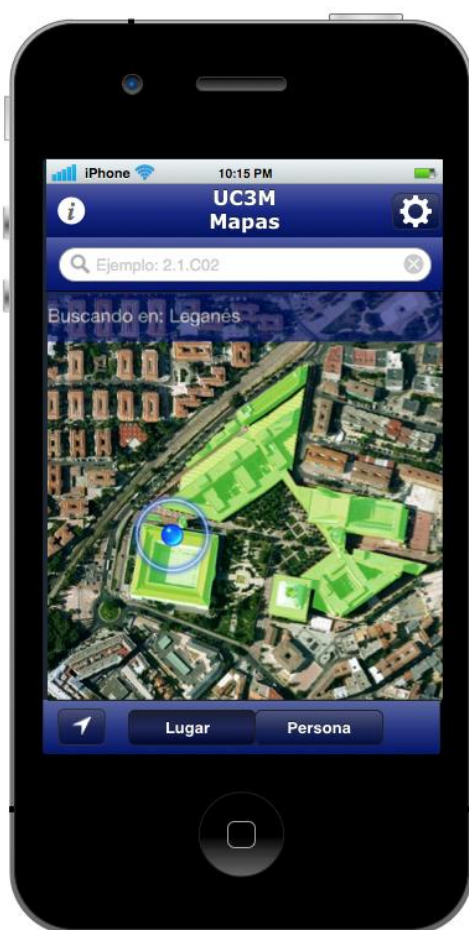


Figura 4.5: Vista principal con búsqueda de lugar seleccionada

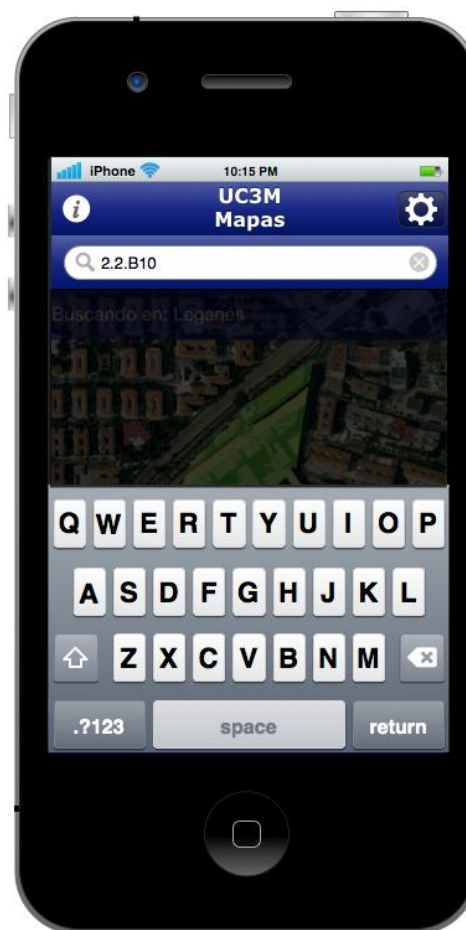


Figura 4.6: Inserción de término de búsqueda de un lugar

En la parte inferior se ha incluido una barra a la que se ha trasladado desde la vista de configuración el botón segmentado que permite elegir el tipo de búsqueda. También contiene un botón que permite al usuario centrar el mapa en su posición actual, teniendo en cuenta el respectivo error de precisión.

Con respecto a la primera versión, el flujo para buscar un lugar es bastante similar. En la vista principal el usuario selecciona el botón de lugar y toca en la barra de búsqueda. Introduce el término de búsqueda como se ve en la Figura 4.6 y toca en la

lupa para ejecutar la búsqueda. La Figura 4.7 refleja qué sucede si para la búsqueda de un lugar, se obtienen varios resultados. La app señala todas las ubicaciones que ha recuperado y complementa la primera recuperada con un globo informativo. El plano de planta que muestra es también el de la primera ubicación recuperada. Como se han obtenido varios resultados, la app no indica qué planta y edificio se están mostrando.

Tras la definición de nuevos requisitos, se estableció mostrar para cada lugar o persona una vista de detalle que aportase más información sobre el resultado de la búsqueda a parte de su ubicación.



Figura 4.7: Resultado de búsqueda de un lugar

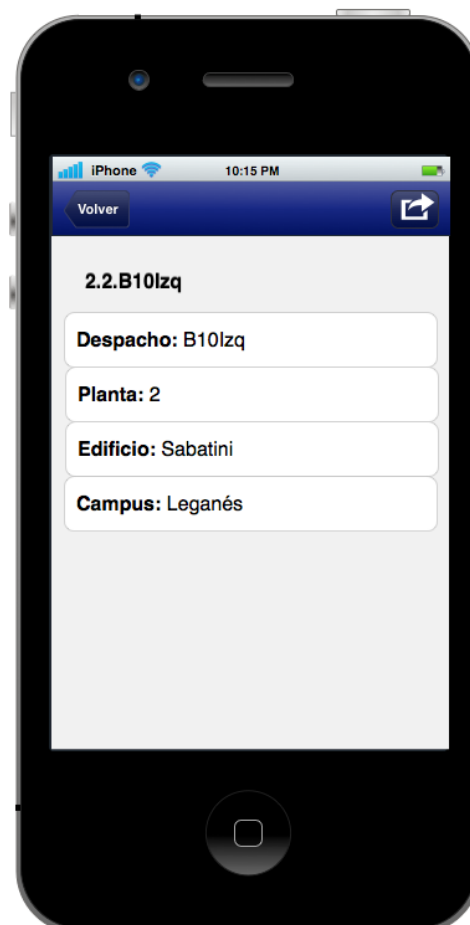


Figura 4.8: Detalle de búsqueda de un lugar

En la Figura 4.8 se ve el detalle para un lugar, consistente en separar las partes del identificador del lugar para aclarar al usuario el edificio, la planta y el despacho a los que se refiere el identificador. En la parte superior se incluye un botón para que el usuario reporte una incidencia en caso de que los datos del lugar sean incorrectos, tanto su ubicación en el plano como los mostrados en la vista de detalle.

Resulta interesante destacar que el plano que finalmente se ofrece al usuario no es el de los paneles indicativos, sino una modificación del plano arquitectónico provisto por la UC3M. Toda la información sobre el tratamiento realizado a los planos se puede encontrar en la sección 4.6.

La parte de cómo el usuario interacciona con la aplicación para realizar una búsqueda de persona se determina entre la Figura 4.9 y la Figura 4.14. Se detalla en esta parte del *paper prototype* el comportamiento de la app en caso de introducir un término de búsqueda común para el nombre de una persona y otro menos común.

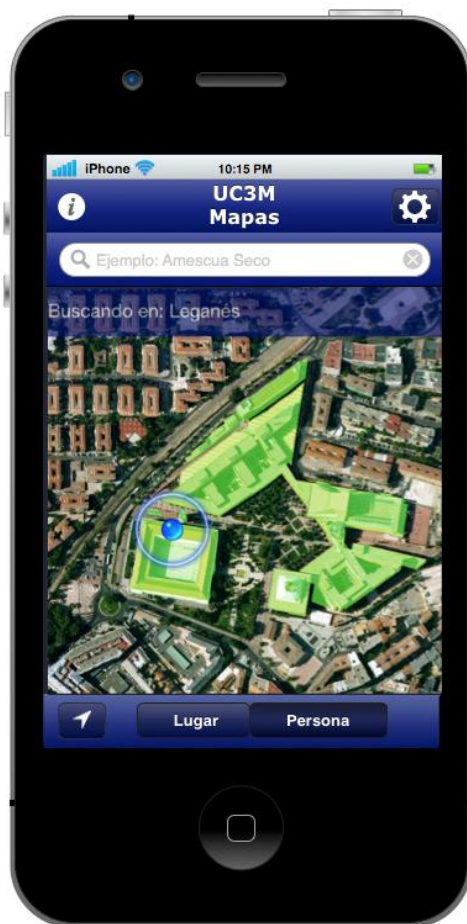


Figura 4.9: Vista principal con búsqueda de persona seleccionada



Figura 4.10: Inserción de término de búsqueda de una persona

Para buscar una persona, el usuario toca el botón persona en la vista principal, Figura 4.9, y después la barra de búsqueda para introducir el término de búsqueda, Figura 4.10. El término introducido como ejemplo es *García*, apellido bastante común en España, por lo que se espera que la app obtenga varios resultados correspondientes a las personas del campus cuyos nombres completos contengan el término de búsqueda usado. La Figura 4.11 enseña cómo la app ofrece al usuario una lista de resultados para que seleccione el que está buscando.



Figura 4.11: Lista de resultados de personas encontradas

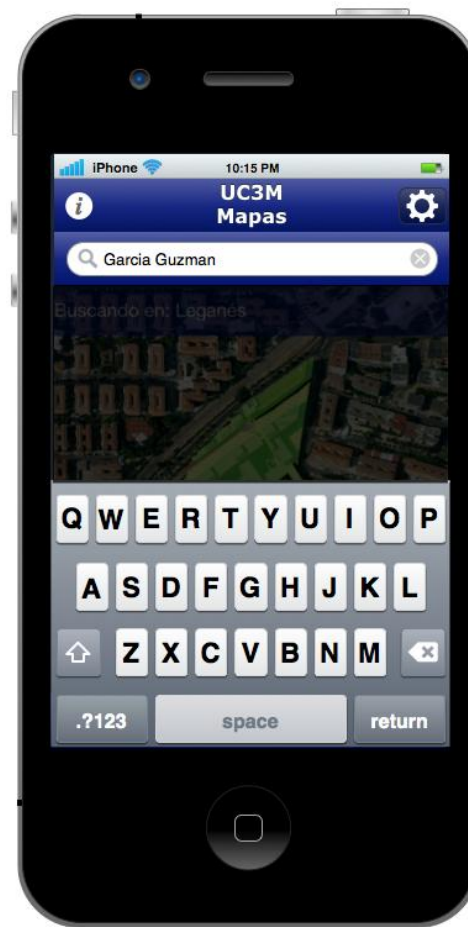


Figura 4.12: Inserción de término de búsqueda de una persona específico

Cuando el resultado de la búsqueda de una persona es único, Figura 4.13, la aplicación indica la ubicación sobre el plano de la planta superpuesto en el mapa e indica el edificio y la planta que se están mostrando. Un ejemplo de término de búsqueda que el usuario puede introducir para obtener un resultado único de persona es *García Guzmán*, Figura 4.12.

Al igual que para el resultado de un lugar, la app complementa la ubicación con un globo informativo que muestra el identificador del lugar. Si el usuario toca sobre el globo informativo, puede acceder a la vista de detalle. La información que se ofrece en la vista de detalle es relativa a la persona que el usuario haya buscado. Se incluye el nombre completo de la persona y el departamento al que pertenece, así como sus datos de contacto en una tabla: correo, despacho, edificio y campus del despacho y el teléfono del despacho. Cuando el usuario toca sobre la fila del correo electrónico, se abre el cliente de correo predeterminado y si toca sobre la fila del teléfono, el dispositivo marca el número de teléfono que contiene. En la parte superior se incluye un botón para notificar incidencias sobre la incorrección de los datos o la falta de información de la persona.



Figura 4.13: Resultado de búsqueda de una persona

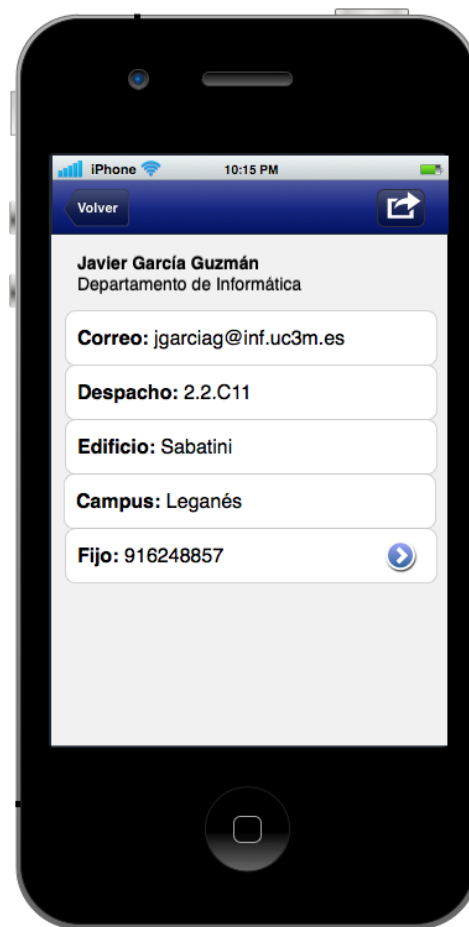


Figura 4.14: Detalle de búsqueda de una persona

Con respecto a la configuración de la app, el usuario puede acceder a ella tocando sobre el botón de configuración de la vista principal. Al igual que en la primera versión, el usuario puede elegir el tipo de mapa y el campus sobre el que realizar las búsquedas. La única diferencia se aprecia en que en esta vista, el usuario ya no puede elegir el tipo de búsqueda porque el botón segmentado se ha trasladado a la vista principal. Cuando el usuario ha finalizado la elección de la configuración, pulsa el botón OK situado en la parte superior. La vista de configuración se puede observar en la Figura 4.15.

Finalmente, el usuario puede tocar sobre el botón de información de la vista principal para acceder a la vista de acerca de. Desde esta vista puede acceder a la ayuda y a los créditos de la app que contienen la versión de la app, una dirección de correo electrónico de contacto, el perfil de Twitter del SEL e información de las partes implicadas en el desarrollo del proyecto.

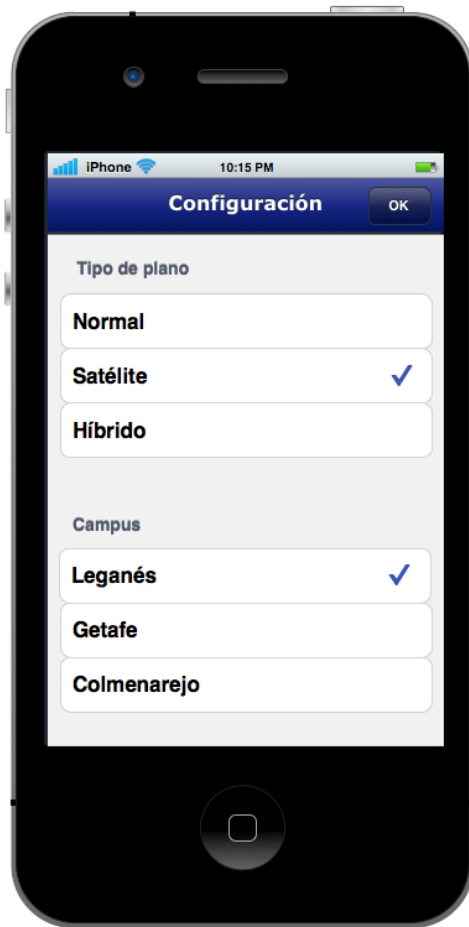


Figura 4.15: Vista de configuración



Figura 4.16: Vista de acerca de

4.2 ALTERNATIVAS DE DISEÑO

En esta sección se explican brevemente las diferentes alternativas que se han barajado durante la fase de diseño de la aplicación y la que se ha elegido para el diseño definitivo.

En las alternativas de diseño se integran diferentes elementos del sistema que también cuentan con diferentes opciones de diseño. Primeramente se va a discutir sobre los elementos del sistema y después, sobre las diferentes alternativas de combinación de los mismos para construir la totalidad del sistema.

4.2.1 ELEMENTOS DEL SISTEMA

Longitud mínima del término de búsqueda: el uso del servicio web proporcionado por la UC3M derivó en un pequeño problema consistente en que si el usuario introducía un término de búsqueda con pocos caracteres, el servicio devolvía gran cantidad de resultados. Tras contactar con el administrador del servicio web, se estableció que el servicio devolviera resultados cuando el término de búsqueda tuviera tres caracteres como mínimo debido a que es la longitud mínima de algunos valores de los campos del directorio.

Servicio web: al inicio del proyecto se tuvo que decidir qué lenguaje se iba a utilizar para devolver los datos que ofrece el servicio web del SEL-UC3M. Las alternativas eran XML y JSON. La primera tiene como ventaja que es muy comprensible y existen muchas herramientas que permiten obtener en el cliente la información codificada. La opción de utilizar JSON tomó fuerza cuando se priorizó la rapidez en las respuestas y por lo tanto, cuanto menor sea el número de bytes enviados, el tiempo de respuesta disminuye. El problema que tiene XML es que su estructura hace que haya que utilizar muchas etiquetas para representar la información, añadiendo peso a los archivos. El formato JSON representa la información de forma más ligera y por ello se estableció su uso para codificar la información devuelta por el servicio web.

Fuente de datos: cuando se diseñó el servicio web del SEL-UC3M, se pensó en recuperar la información de personas directamente del directorio LDAP de la UC3M. Esta opción finalmente se descartó puesto que para este fin, es necesario utilizar un usuario y una contraseña con permiso para acceder al directorio LDAP. El acceso se realiza con una cuenta personal y por razones de seguridad, no es recomendable incluirla en la app. Por este motivo, se decidió comunicar el servicio web del SEL-UC3M con un servicio web ofrecido por la UC3M, de manera que la app no acceda al directorio de personas directamente.

Formato de los datos devueltos: a nivel interno se tuvo que definir cómo debía devolver los datos el servicio web del SEL-UC3M cuando hubiera personas con más de un despacho. Se barajó la posibilidad de devolver dos objetos JSON de tipo persona, uno por cada localización, pero como consecuencia se recibían datos por duplicado como el correo electrónico, el nombre, los apellidos y el departamento en el que trabaja la persona buscada.

Finalmente, la función encargada de recuperar la información de las personas, denominada *person2*, devuelve la información con el siguiente formato:

```
[
  {
    "uid": 3840,
    "name": "Maria Isabel",
    "lastname": "Barro Vivero",
    "department": "Ingenieria Telematica",
    "email": "ibarro@di.uc3m.es",
    "locations": [
      {
        "roomNumber": "1.1.J05",
        "building": "Agustin De Betancourt",
        "campus": "Leganés",
        "telephone": "916249992"
      },
      {
        "roomNumber": "4.1.F15",
        "building": "Torres Quevedo",
        "campus": "Leganés",
        "telephone": "916249992"
      }
    ]
  }
]
```

Como se puede ver, retorna un solo objeto JSON que contiene todos los lugares que una persona tiene asignados en un campus. Durante el proceso de diseño también se definió el formato de objeto que contiene la información de un lugar. Como campos, resulta imprescindible que contenga los datos del identificador y las coordenadas geográficas donde se encuentra ubicado. La función encargada de devolver la información se denomina *find* y retorna objetos con la siguiente estructura:

```
[
  {
    "id_place": "240",
    "building": "Al Aire Libre",
    "buildingNumber": "0",
    "floor": "0",
    "lat": "40.332695",
    "lng": "-3.767110",
    "name": "Locomotora",
    "isRoom": "0"
  }
]
```

Autocompletado: durante el desarrollo de la aplicación se contempló como alternativa de diseño incluir la funcionalidad de autocompletado en la barra de búsqueda a semejanza de la app Google Maps. En esta aplicación, mientras el usuario escribe el término de búsqueda, se le va ofreciendo distintas sugerencias para darle facilidad a la hora de encontrar lo que busca sin necesidad de escribir el término de búsqueda completo.

El problema que se encontró cuando se hicieron las primeras pruebas en UC3M Mapas fue que, cuando se realizaba una búsqueda de persona y se escribía el tercer carácter del término de búsqueda, la app realizaba una consulta al servicio web para comenzar a ofrecer sugerencias con el inconveniente de que el servicio web devolvía cien resultados en muchos de los casos, viéndose la aplicación ralentizada.

La implementación del servicio web de la UC3M hace posible introducir un término de búsqueda referente a cualquier dato de la persona, motivo por el cual se obtienen tantos resultados siendo cien el máximo establecido por el administrador del servicio. Se hizo otra prueba estableciendo en cuatro y cinco caracteres el número mínimo de caracteres escritos por el usuario para ofrecer sugerencias y de nuevo surgieron problemas de optimización: si un usuario se equivocaba y modificaba un carácter, se aumentaba el número de consultas y equivocándose repetidas veces en el mismo término, provocaba de nuevo la ralentización de la app.

Teniendo en cuenta los problemas surgidos, se decidió no implementar la funcionalidad de autocompletado ya que los inconvenientes aparecidos eran mayores que los beneficios que se perseguía aportar.

Fotografías de personas: durante el diseño se pensó que sería interesante poder ver en la vista de detalle de persona la fotografía de la persona buscada ya que para algunas personas es más sencillo recordar a una persona por su cara que por su nombre.

Esta característica se llegó a implementar pero se terminó descartando porque al no requerir la aplicación de inicio de sesión, se infringiría la Ley Orgánica de Protección de Datos (LOPD).

4.2.2 PRIMERA ALTERNATIVA DE DISEÑO

La primera alternativa consiste en que el servicio web del SEL-UC3M accede directamente a la base de datos de lugares y al repositorio de planos, ambos gestionados por el SEL-UC3M, y al LDAP de la UC3M para obtener los datos de las personas.

En esta alternativa se maneja el uso del autocompletado a partir de cinco caracteres para que la aplicación no se vea ralentizada ya que a partir de esa cifra no se esperan demasiadas sugerencias de búsqueda.

En esta alternativa la aplicación realiza las consultas necesarias al servicio web del SEL-UC3M y este devuelve la información en formato JSON.

En la Figura 4.17 se puede ver un esquema del diseño planteado en esta alternativa.

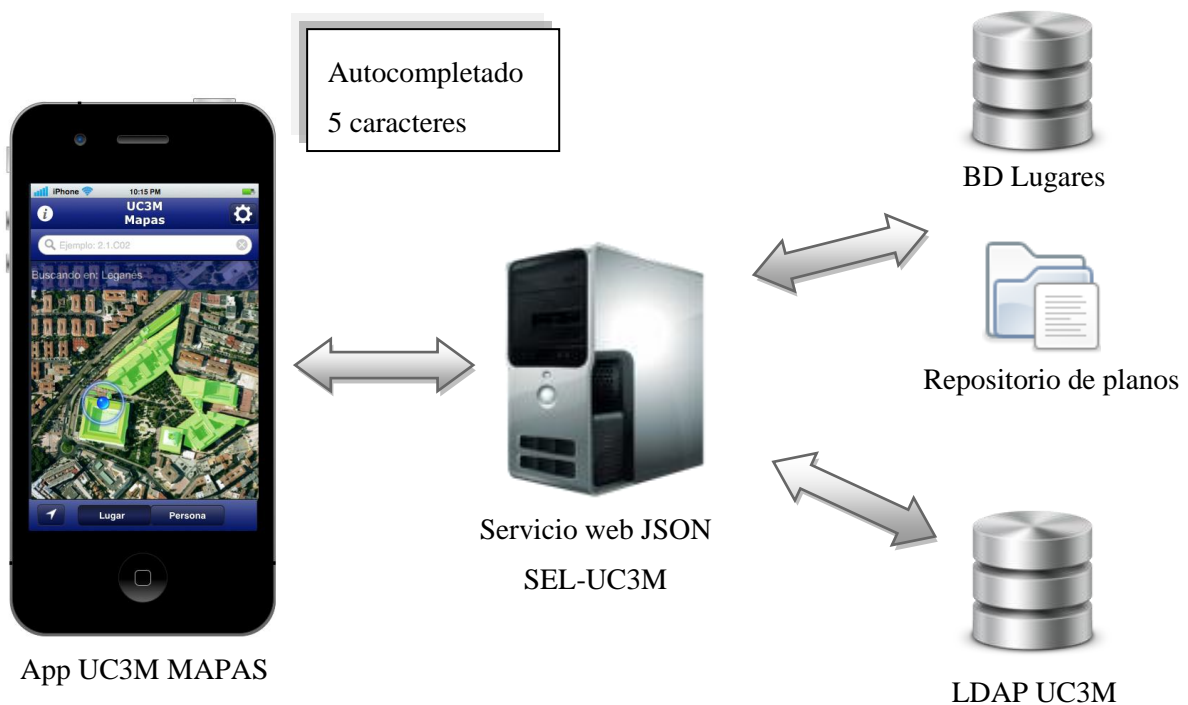


Figura 4.17: Primera alternativa de diseño

La ventaja de esta alternativa es la velocidad, pero tiene dos inconvenientes:

1. La información que devuelve LDAP no tiene una estructura comprensible y por ello, es necesario realizar un procesamiento previo en el servicio web del SEL-UC3M para que este los pueda mandar a la aplicación en un formato manejable.
2. Es necesario que el servicio web del SEL-UC3M se autentique en LDAP para realizar consultas.

4.2.3 SEGUNDA ALTERNATIVA DE DISEÑO

La segunda alternativa incluye, con respecto a la primera, un servicio web desarrollado por la UC3M, en concreto por el Servicio de Informática y Comunicaciones, que se encarga de hacer las consultas sobre el LDAP de la UC3M y proporcionar en formato JSON los resultados al servicio web del SEL-UC3M. La información es menos costosa de procesar por el servicio web del SEL-UC3M con respecto a la primera alternativa planteada.

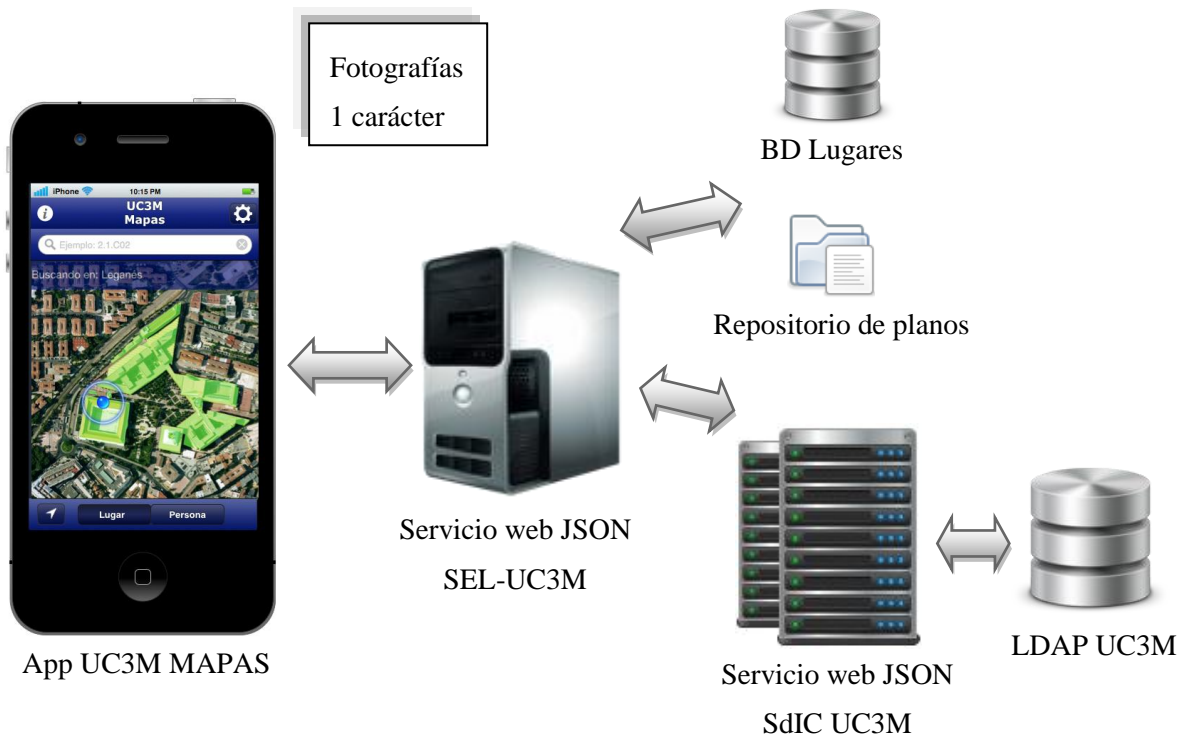


Figura 4.18: Segunda alternativa de diseño

Al introducir un salto más entre la aplicación y el LDAP de la UC3M, las consultas sobre personas se vuelven más lentas y añadido al autocompletado, ralentiza de forma muy perceptible por el usuario el funcionamiento de la app. Se decide en esta alternativa eliminar el autocompletado y permitir que la aplicación inicie consultas para búsquedas con términos de búsqueda de un carácter. Será necesario para la búsqueda de personas introducir tres términos de búsqueda por el diseño del servicio de la UC3M.

Además, en esta alternativa se presenta la opción de incluir fotografías de las personas en la vista de detalle que proporciona la aplicación. Esta opción se tiene en cuenta porque el servicio web proporcionado por el Servicio de Informática y Comunicaciones devuelve, entre otros datos, la URL de la fotografía de la persona.

Del mismo modo que en la primera alternativa, la app solicita al servicio web del SEL-UC3M las consultas sobre las bases de datos de lugares y personas. El servicio web del SEL-UC3M accede directamente a la base de datos de lugares y como se ha explicado, a través de un servicio web al directorio LDAP. La información solicitada es mandada a la app en formato JSON. El esquema completo de la segunda alternativa puede observarse en la Figura 4.18.

4.2.4 TERCERA ALTERNATIVA DE DISEÑO

La tercera alternativa de diseño es igual que la segunda alternativa a excepción de que se ha decidido dejar de incluir la fotografía de una persona buscada. El hecho de que usuarios de la app externos a la UC3M puedan acceder a las fotografías del personal de la UC3M junto a sus datos personales infringe la Ley Orgánica de Protección de Datos. En caso de que en un futuro se quiera incluir la fotografía de la persona buscada, se barajarían alternativas como restringir el acceso a los datos y fotografía de las personas a usuarios que pertenezcan a la comunidad universitaria.

La tercera alternativa de diseño es la que se ha elegido para el sistema en el que está incluido el proyecto. Puede verse en la Figura 4.19.

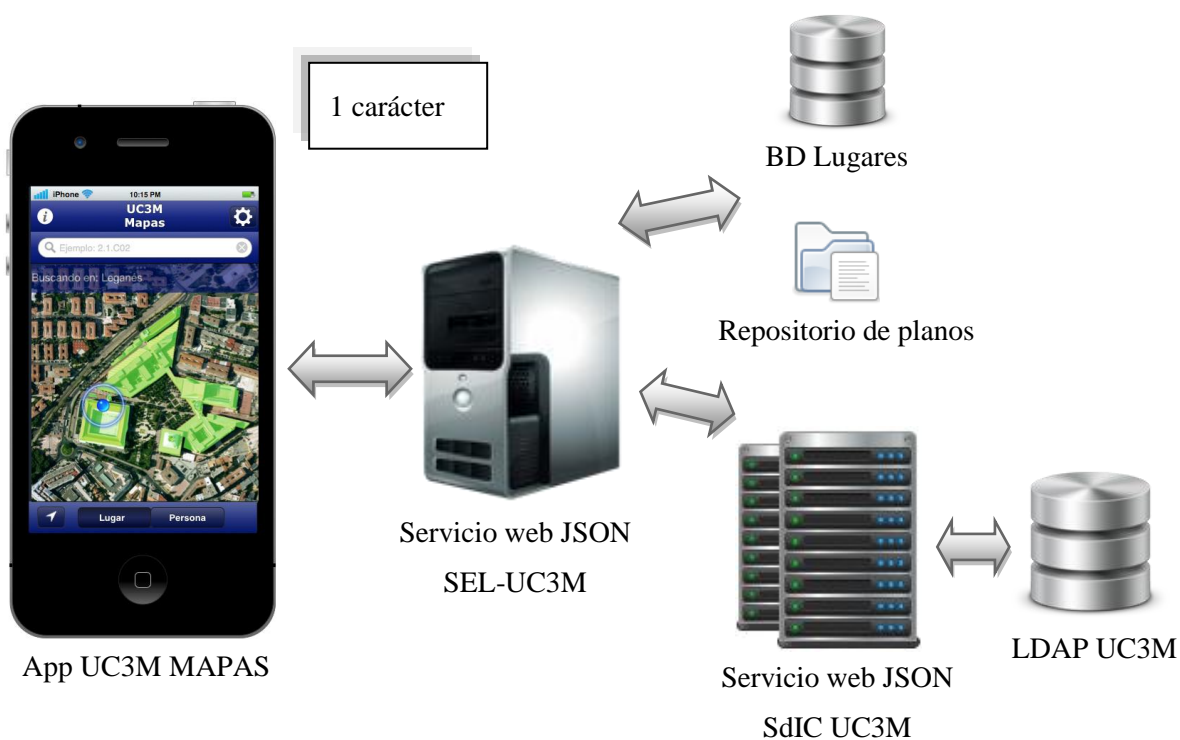


Figura 4.19: Tercera alternativa de diseño

4.3 DIAGRAMAS DE COMPONENTES Y DE CLASES

Una vez elegida la alternativa de diseño del sistema a desarrollar, ver Figura 4.19, se ha modelado la aplicación utilizando diagramas UML.

Los diagramas utilizados en esta sección son el diagrama de componentes y el diagrama de clases. El diagrama de componentes modela la arquitectura software de la aplicación, agrupando diferentes partes de la aplicación y mostrando la relación proveedor – consumidor de servicios entre componentes. El diagrama de clases muestra la composición de las clases y la relación entre ellas.

La Figura 4.20 muestra el diagrama de componentes de la aplicación. Se pueden distinguir los siguientes componentes:

Controlador: es un paquete que agrupa a tres componentes que incluyen diferentes elementos que implementan la lógica de la aplicación. Ofrece el puerto *UI_Sender* por el que las interfaces de usuario le mandan su estado actual a los componentes Map y Options y el puerto *Data* por donde el componente Map recibe la información solicitada a los recursos web.

Map: está incluido en el paquete Controlador. Incluye las clases que dan soporte directo a la funcionalidad principal de la aplicación. Estas clases son las encargadas de controlar el mapa cartográfico, superponer los planos e indicar ubicaciones.

Se comunica con los siguientes componentes:

- Support, para utilizar los recursos de *frameworks* y librerías de terceros que posibilitan implementar la funcionalidad principal.
- Options, para mandarle información de inicialización a las clases que componen este paquete y recibir información de finalización de las mismas.
- User Interface, para mandar y recibir información de control sobre las interfaces de usuario correspondientes de la aplicación.
- Paquete Server, para recuperar información de lugares y personas y las *tiles* de los planos de las plantas.

Ofrece el puerto *M_Receiver* para recibir información de finalización de las clases del componente Options.

Options: está incluido en el paquete Controlador. Incluye las clases que dan soporte a las funcionalidades secundarias de la aplicación. Estas clases son las encargadas de implementar la lógica para mostrar los detalles de un lugar y de una persona, la configuración de la aplicación y los créditos y la ayuda de la misma.

Se comunica con los siguientes componentes:

- Support, para utilizar los recursos de *frameworks* y librerías de terceros que posibilitan implementar las funcionalidades secundarias.
- Map, para mandarle información de finalización de las clases y recibir información de inicialización de las mismas.

- User Interface, para mandar y recibir información de control sobre las interfaces de usuario correspondientes de la aplicación.

Ofrece el puerto *O_Receiver* para recibir información de inicialización de sus clases.

Support: está incluido en el paquete Controlador. Incluye los *frameworks* del sistema operativo y las librerías desarrolladas por terceros que dan soporte a la implementación de las funcionalidades de la aplicación. Ofrece el puerto Utilities para proveer a los componentes que lo necesiten los servicios que contiene.

User Interface: incluye las clases que implementan las interfaces de usuario. Se comunica con los componentes Map y Options para recibir y enviar información de control sobre el estado actual de las interfaces. Ofrece el puerto UI_Receiver para que los componentes del paquete Controlador manden órdenes de control para actualizar o recuperar el estado actual de las interfaces.

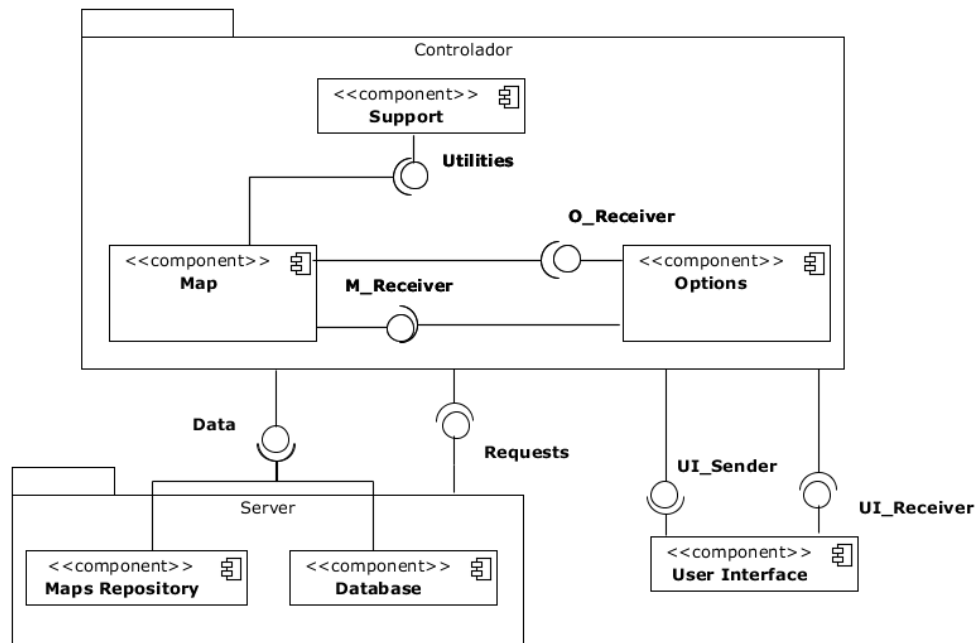


Figura 4.20: Diagrama de componentes de la aplicación

Server: es un paquete que agrupa a los componentes que incluyen la base de datos de lugares georreferenciada y el directorio de personas y las *tiles* de los planos de las plantas de los edificios de la UC3M. Ofrece el puerto Requests para que los componentes del paquete Controlador le realicen las consultas que necesiten.

Database: está incluido en el paquete Server. Incluye los recursos web a los que la aplicación accede para obtener información sobre lugares y personas de la UC3M. Se comunica con los componentes del paquete Controlador para proporcionarles la información que le soliciten a través del paquete Server.

Maps Repository: está incluido en el paquete Server. Incluye los recursos web a los que la aplicación accede para obtener las imágenes de las *tiles* de los planos que

necesita. Se comunica con los componentes del paquete Controlador para proporcionarles la información que le soliciten a través del paquete Server.

Tras haber comprendido el diagrama de componentes, se puede deducir que la aplicación se ha diseñado siguiendo el patrón de diseño Modelo-Vista-Controlador o, por sus siglas, MVC. Este patrón propone un controlador encargado de recoger las interacciones del usuario en las vistas y actualizar las vistas en función de las acciones que desencadenen. El controlador también interactúa con el modelo el cual, le proporciona la información y los datos necesarios para llevar las acciones a cabo. Se puede encontrar más información y ejemplos prácticos de cómo se implementa el MVC en las aplicaciones para iOS en el capítulo 6 de [15].

Conocido el diagrama de componentes se va a pasar a explicar las clases que incluyen los componentes.

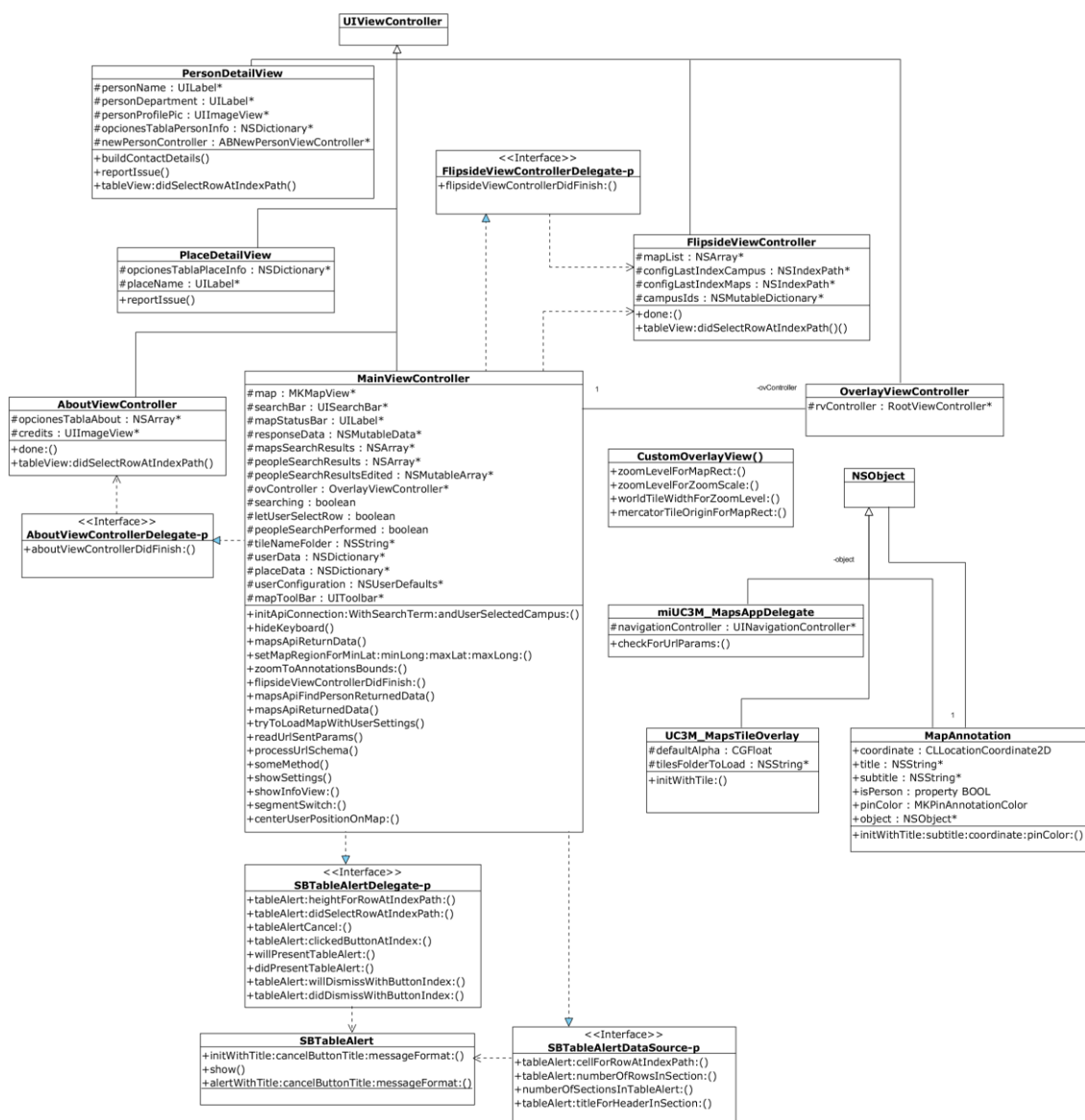


Figura 4.21: Diagrama de clases de la aplicación

Para mostrar las relaciones entre las clases se ha realizado el diagrama de clases que se puede ver en la Figura 4.21. El diagrama de clases únicamente incluye las clases contenidas en los paquetes Maps y Options ya que son los que implementan la funcionalidad de la app iOS. También forman parte de la app las clases de los componentes User Interface y Support y se ha decidido no incluir sus clases en el diagrama porque para el primero, las clases que implementan las interfaces de usuario son realmente ficheros escritos en lenguaje de marcado XML y para el segundo, las clases han sido implementadas por terceras personas al tratarse de *frameworks* y librerías.

Con la finalidad de poder observar el diagrama de clases con detalle, este se ha separado en función del componente al que pertenece cada clase.

Clases del componente Map

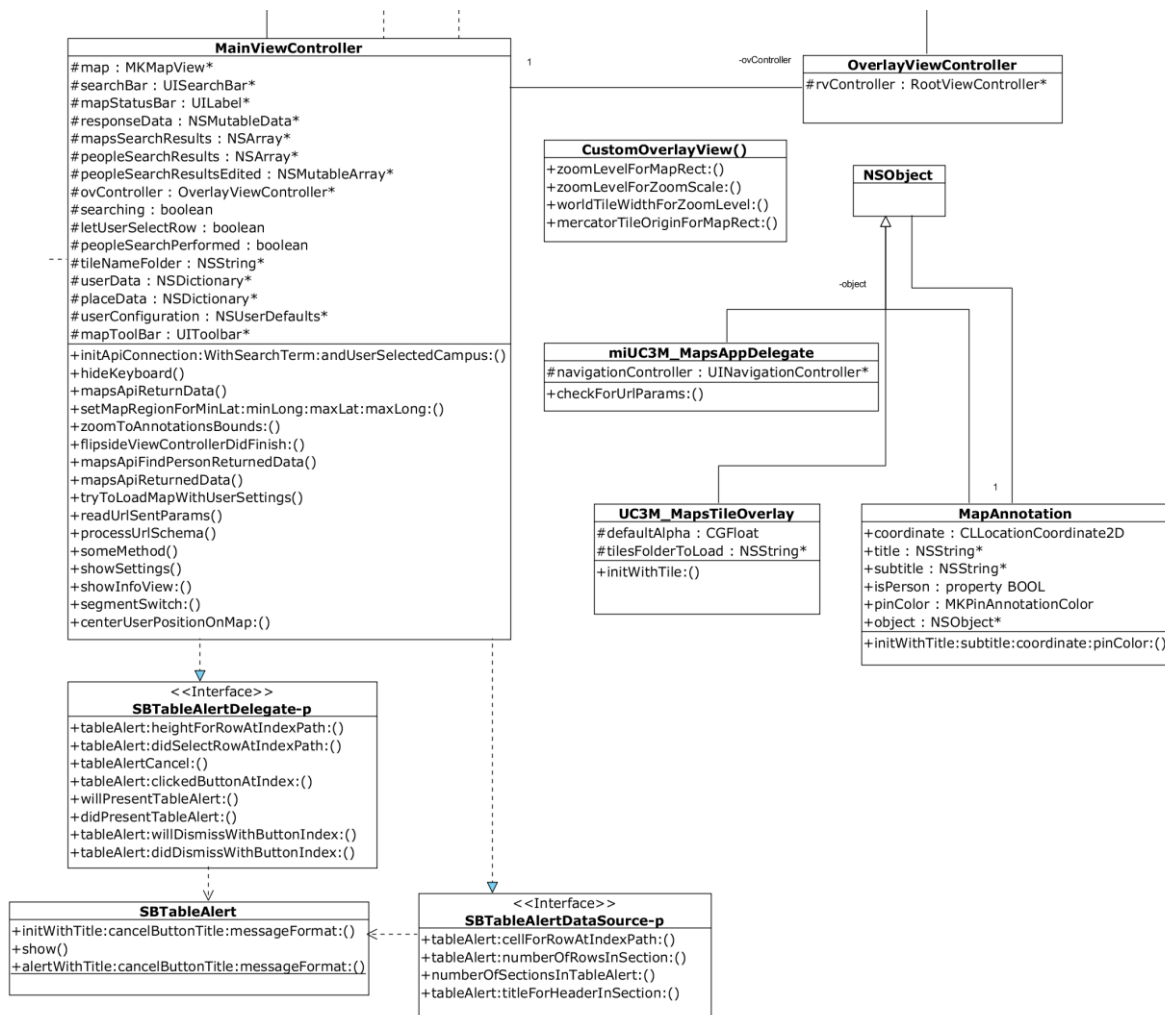


Figura 4.22: Clases del componente Map

Las clases que están incluidas en el componente Map se pueden ver en la Figura 4.22.

La clase *MainViewController* implementa la mayor parte de la lógica de la aplicación ya que maneja la vista principal de la interfaz y ofrece la funcionalidad principal. Como atributos, se definen los conectores que permiten controlar los elementos de la vista principal, como la vista del mapa, la barra de búsqueda, el indicador del plano que se está mostrando o las barras de herramienta. También contempla atributos para manejar información obtenida de los servicios web. Los métodos que implementa permiten realizar, principalmente, las consultas a los servicios web y el tratamiento de la información que retornan, manejar los eventos de la interfaz de usuario de la vista principal, mostrar el mapa cartográfico y superponerle los planos y la indicación de las ubicaciones, y por último, iniciar las clases que controlan la visualización de los detalles de las búsquedas y la configuración, ayuda y créditos de la aplicación.

Las clases *UC3M_MapsTileOverlay* y *CustomOverlayView* se encargan de facilitar a la clase *MainViewController* mostrar los planos en la aplicación. En especial, *UC3M_MapsTileOverlay* se encarga de realizar las peticiones de las tiles, o porciones, de los planos al servicio web del SEL-UC3M.

La clase *MapAnnotation* representa un indicador de ubicación que se coloca sobre el mapa y un plano para señalar la ubicación del resultado de una búsqueda. Como atributos incluye las coordenadas de dónde tiene que ser posicionado, el nombre del lugar y el color en el que tiene que ser pintado por la vista principal.

Las clases *SBTableAlert*, *SBTableAlertDelegate* y *SBTableAlertDataSource* se encargan de facilitar a la clase *MainViewController* mostrar una tabla de resultados superpuesta en la vista principal.

La clase *miUC3M_MapsAppDelegate* es la que maneja el inicio de la aplicación.

Clases del componente Options

Las clases que están incluidas en el componente Options se pueden ver en la Figura 4.23.

La clase *PersonDetailView* se encarga de manejar la vista de detalle de una persona buscada. Contiene los atributos que actúan como conectores con los elementos que muestran la información detallada de una persona en la vista correspondiente de la interfaz de usuario. Los atributos proporcionan información a la vista sobre el nombre completo y el departamento de la persona y sus datos de contacto. Contiene un método encargado de detectar y realizar la acción correspondiente en caso de que el usuario elija mandar un correo a la persona, llamarla por teléfono, agregarla como contacto del teléfono o mandar una incidencia al administrador de la app. También contiene un método para mandar una incidencia al administrador de la app.

La clase *PlaceDetailView* se encarga de manejar la vista de detalle de un lugar buscado. Contiene los atributos que actúan como conectores con los elementos que muestran la información detallada de un lugar en la vista correspondiente de la interfaz de usuario. Los atributos proporcionan información a la vista sobre el identificador del lugar y el significado de los campos del mismo. Contiene un método para mandar una incidencia al administrador de la app.

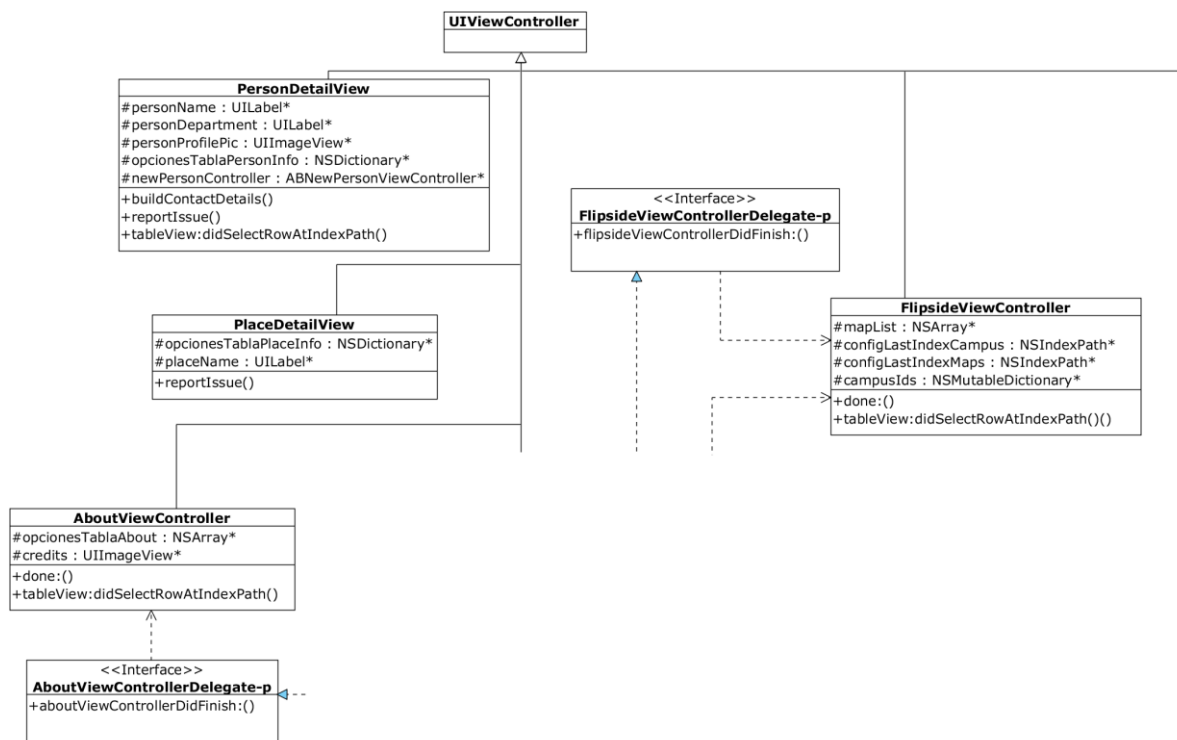


Figura 4.23: Clases del componente Options

La clase *FlipsideViewController* se encarga de manejar la vista de configuración de la aplicación. Contiene los atributos que actúan como conectores con los elementos que muestran, en la vista correspondiente de la interfaz de usuario, la información de los tipos de mapas y campus sobre los que se puede elegir realizar las búsquedas. Los atributos proporcionan información a la vista sobre los tipos de mapas y campus que están disponibles. Contiene un método encargado de detectar y realizar la acción correspondiente cuando el usuario selecciona una opción de configuración.

La clase *AboutViewController* se encarga de manejar la vista de créditos y ayuda de la aplicación. Contiene los atributos que actúan como conectores con los elementos que muestran, en la vista correspondiente de la interfaz de usuario, los datos relevantes de la aplicación. Los atributos proporcionan información a la vista sobre los datos de interés disponibles. Contiene un método que se encarga de detectar y realizar la acción correspondiente en función de la opción elegida por el usuario.

4.4 DIAGRAMAS DE SECUENCIA

Una vez realizado el diagrama de clases en el que se muestran las relaciones entre las clases de cada uno de los componentes del sistema, se procede a definir los diagramas de secuencia que sirven al programador como modelo a la hora de desarrollar la interacción entre los objetos de la aplicación.

El primero de ellos se expone en la Figura 4.24 y muestra los métodos que se ejecutan al iniciar la aplicación UC3M Mapas. Primeramente, el método *viewDidLoad* construye la vista principal y después, el método *viewDidAppear* intenta mostrar el mapa cartográfico con las preferencias de tipo de mapa y campus almacenadas.

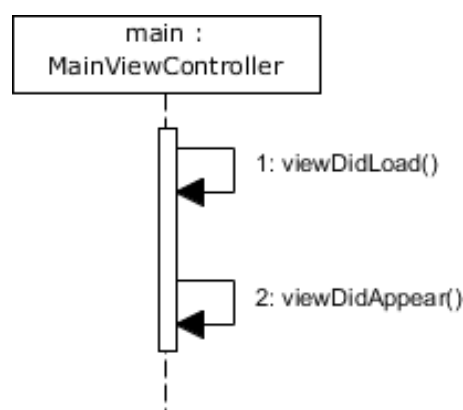


Figura 4.24: Diagrama de secuencia de inicio de la aplicación

El segundo diagrama de secuencia muestra los métodos que la aplicación realiza internamente cuando el usuario efectúa una búsqueda de lugar. Los pasos, mostrados en la Figura 4.25, desde que el usuario ha introducido el término de búsqueda hasta que la aplicación retorna de la vista detalle de lugar, son los siguientes:

1. Forma la URL de petición al servicio web del SEL-UC3M a partir de los parámetros recibidos por el método que se encarga de construirla. En estos se indica si la búsqueda es de lugar o de persona, el término de búsqueda y el campus sobre el que se quiere realizar la búsqueda.
2. Inicia la conexión con el servicio web del SEL-UC3M.
3. Recibe los datos de respuesta que genera la consulta.
4. Limpia el mapa para quitar resultados de anteriores búsquedas y como la búsqueda activada es de lugares, llama al método que sabe gestionar el resultado recibido para búsquedas de lugares. Este método se encarga de indicar sobre el mapa la ubicación del lugar o lugares, centrar el mapa en la ubicación del lugar de forma que se visualicen todos los resultados, cargar las tiles del plano de la planta del edificio al que corresponde el lugar e indicar el plano que se está mostrando.

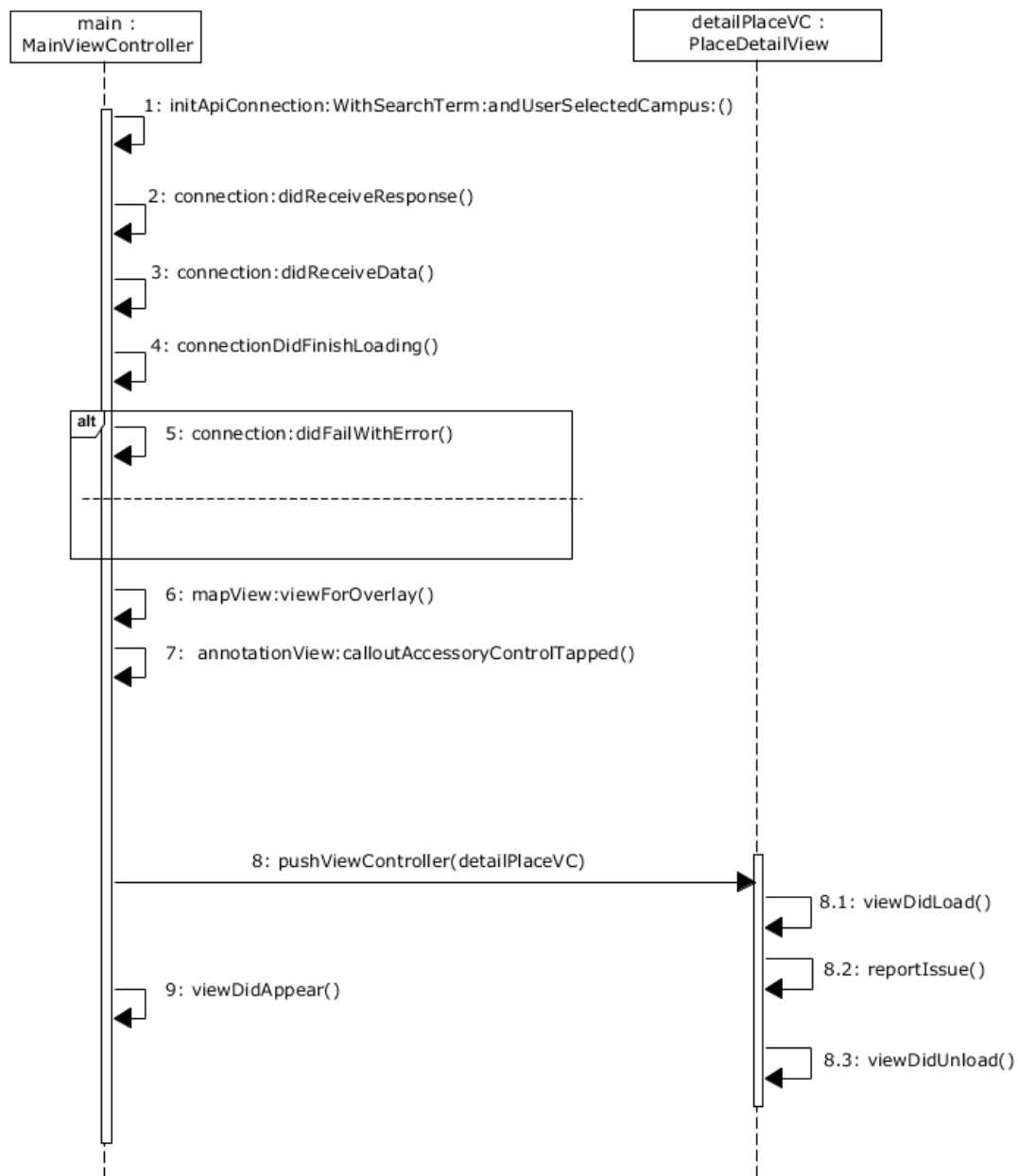


Figura 4.25: Diagrama de secuencia de búsqueda de un lugar

5. En caso de que la conexión no se realice con éxito, muestra un mensaje de aviso al usuario.
6. Muestra las tiles del plano de la planta en el mapa.
7. Detecta que el usuario ha seleccionado el globo de texto que aparece en la ubicación del lugar y manda llamar al método que inicia la vista de detalle del lugar.

8. Inicia la vista de detalle de lugar a la que le proporciona los datos necesarios para que muestre la información del mismo.
 - 8.1. Recoge la información del lugar y construye la vista detalle.
 - 8.2. Detecta que el usuario ha seleccionado la opción de reportar incidencia e inicia la aplicación de correo electrónico.
 - 8.3. Vuelve de la aplicación de correo electrónico y el usuario sale de la vista de detalle de lugar. La vista realiza las acciones correspondientes antes de cerrarse.
9. Cuando vuelve a la vista principal la recarga en el mismo estado en el que estaba antes de ir a la vista de detalle de lugar.

El tercer diagrama de secuencia ofrece la sucesión de pasos que realiza la aplicación cuando el usuario efectúa una búsqueda de persona.

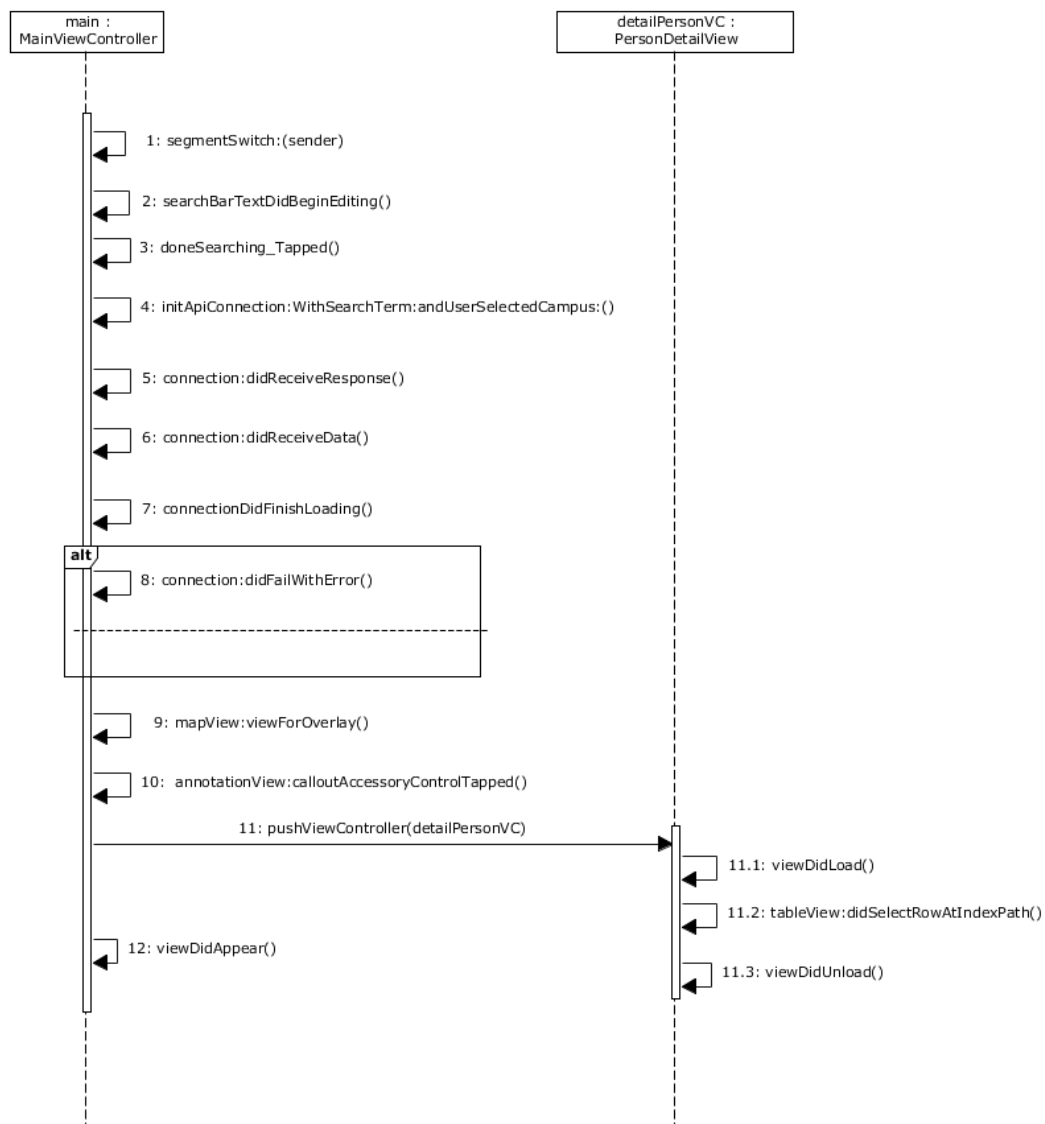


Figura 4.26: Diagrama de secuencia de búsqueda de una persona

Los pasos que se detallan a continuación se pueden ver en la Figura 4.26 y explican la secuencia para la búsqueda de una persona desde que el usuario cambia a la búsqueda de persona, hasta que la aplicación regresa desde la vista detalle de persona.

1. La aplicación detecta que el usuario ha cambiado el tipo de búsqueda a búsqueda de personas.
2. La aplicación captura el término de búsqueda que el usuario introduce.
3. La aplicación detecta que el usuario acaba de finalizar de escribir el término de búsqueda.
4. Forma la URL de petición al servicio web del SEL-UC3M a partir de los parámetros recibidos por el método que se encarga de construirla. En estos se indica si la búsqueda es de lugar o de persona, el término de búsqueda y el campus sobre el que se quiere realizar la búsqueda.
5. Inicia la conexión con el servicio web del SEL-UC3M.
6. Recibe los datos de respuesta que genera la consulta.
7. Limpia el mapa para quitar resultados de anteriores búsquedas y como la búsqueda activada es de personas, llama al método que sabe gestionar el resultado recibido para búsquedas de personas. Este método se encarga de indicar sobre el mapa la ubicación de la persona, centrar el mapa en la ubicación de la persona, cargar las tiles del plano de la planta del edificio al que corresponde la ubicación del lugar de la persona e indicar el plano que se está mostrando. En caso de que se haya encontrado más de una persona, únicamente muestra sobre la vista principal una tabla con los resultados de la búsqueda y cuando el usuario selecciona uno, la aplicación realiza el resto de acciones.
8. En caso de que la conexión no se realice con éxito, muestra un mensaje de aviso al usuario.
9. Muestra las tiles del plano de la planta en el mapa.
10. Detecta que el usuario ha seleccionado el globo de texto que aparece en la ubicación de la persona y manda llamar al método que inicia la vista de detalle de persona.
11. Inicia la vista de detalle de persona a la que le proporciona los datos necesarios para que muestre la información de la misma.
 - 11.1. Recoge la información de la persona y construye la vista detalle.
 - 11.2. Detecta que el usuario ha seleccionado una opción de entre las disponibles en la tabla de datos de contacto de la persona y ejecuta la acción correspondiente.
 - 11.3. El usuario sale de la vista de detalle de persona y realiza las acciones correspondientes antes de cerrarse.
12. Cuando vuelve a la vista principal la recarga en el mismo estado en el que estaba antes de ir a la vista de detalle de persona.

El cuarto diagrama de secuencia realizado muestra la secuencia de pasos que realiza la aplicación cuando el usuario inicia la vista de configuración.

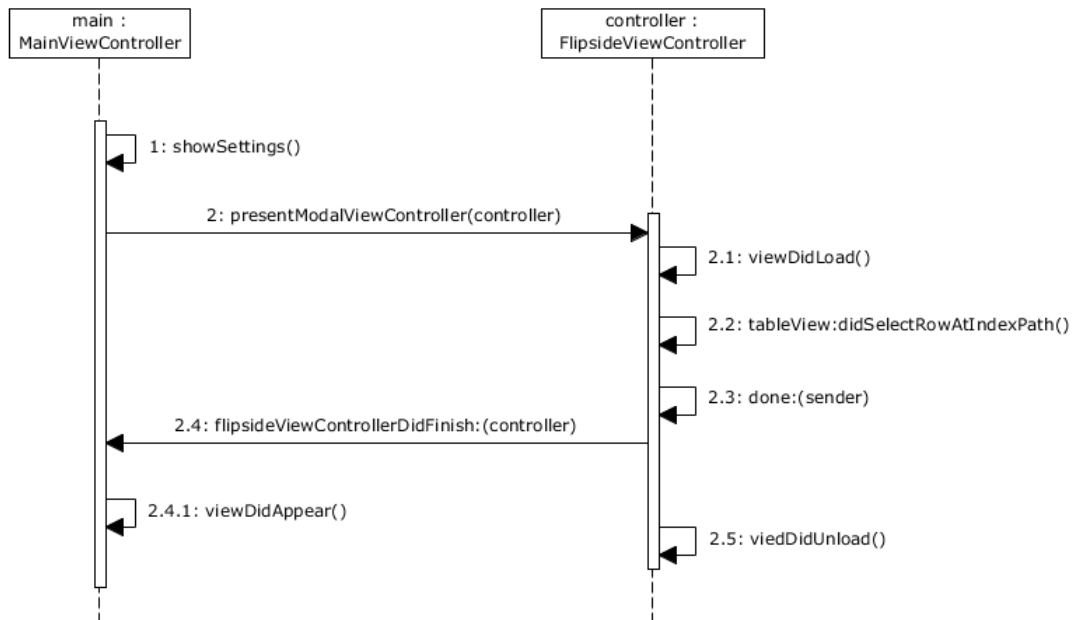


Figura 4.27: Diagrama de secuencia de configuración

En la Figura 4.27 la secuencia de métodos que la aplicación utiliza desde que el usuario selecciona la opción de configuración hasta que vuelve a la vista principal después de haber cambiado alguna opción de configuración. La secuencia completa es la siguiente:

1. La aplicación detecta que el usuario ha seleccionado la opción de configuración en la vista principal.
2. Se ejecuta el método que cambia a la vista de configuración.
 - 2.1. La aplicación crea la vista de configuración con los datos que tiene almacenados de la última configuración realizada por el usuario. Estos datos serán los que la aplicación almacena por defecto si se trata de la primera vez que el usuario configura la app.
 - 2.2. La aplicación detecta que el usuario ha seleccionado una opción de configuración y realiza la acción de actualización correspondiente sobre los datos de configuración de la app.
 - 2.3. La aplicación detecta que el usuario sale de la vista de configuración.
 - 2.4. La vista de configuración devuelve el control a la vista principal.
 - 2.4.1. La vista principal se recarga en el mismo estado en el que estaba antes de ir a la vista configuración.
 - 2.5. La vista de configuración realiza las acciones correspondientes antes de cerrarse.

Por último se muestra el diagrama de secuencia correspondiente a la vista de acerca de, en la que se ve la secuencia de pasos que la aplicación realiza cuando el usuario visualiza la información de la app.

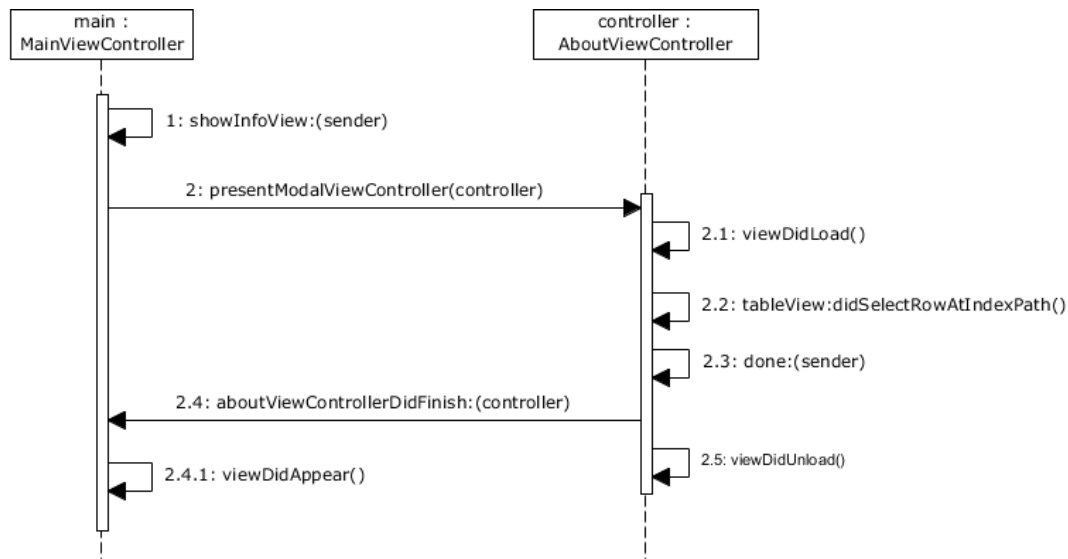


Figura 4.28: Diagrama de secuencia de acerca de

La Figura 4.28 ofrece los métodos que ejecuta la aplicación desde que el usuario accede a la vista de acerca de hasta que sale de la misma. La secuencia de pasos que sigue es:

1. La aplicación detecta que el usuario ha seleccionado en la vista principal la opción de visualizar los créditos y la ayuda.
2. Se ejecuta el método que cambia a la vista de acerca de.
 - 2.1. La aplicación crea la vista de acerca con los datos de los créditos de la ayuda que tiene almacenados.
 - 2.2. La aplicación detecta que el usuario ha seleccionado una opción de entre las disponibles y realiza la correspondiente.
 - 2.3. La aplicación detecta que el usuario sale de la vista de acerca de.
 - 2.4. La vista de acerca de devuelve el control a la vista principal.
 - 2.4.1. La vista principal se recarga en el mismo estado en el que estaba antes de ir a la vista de acerca de.
 - 2.5. La vista de configuración realiza las acciones correspondientes antes de cerrarse.

Los diagramas de secuencia que se han explicado corresponden a las interacciones más comunes entre objetos de las clases de la aplicación ya que instancian aquellas que proveen las principales funcionalidades.

4.5 DISEÑO DE LA BASE DE DATOS

En el presente documento se ha explicado que la app realiza peticiones a un servicio web proporcionado por el grupo SEL-UC3M para obtener información sobre un lugar o una persona buscada por el usuario. También se ha mostrado de manera implícita que el servicio web realiza las búsquedas de lugares sobre una base de datos que contiene la georreferenciación de todos los lugares de los campus de la UC3M. Se ha considerado incluir el diseño de la citada base de datos, en el cual el autor del proyecto ha colaborado, para ofrecer al lector mayor visión del sistema y para que le sea más sencillo comprender su conjunto. Esta base de datos se encuentra implementada en el componente Database del diagrama de componentes del sistema.

En la Figura 4.29 se muestra el modelo entidad – relación de la base de datos. Se presentan las diferentes entidades de la base de datos con sus respectivos atributos y las relaciones que existen entre ellas.

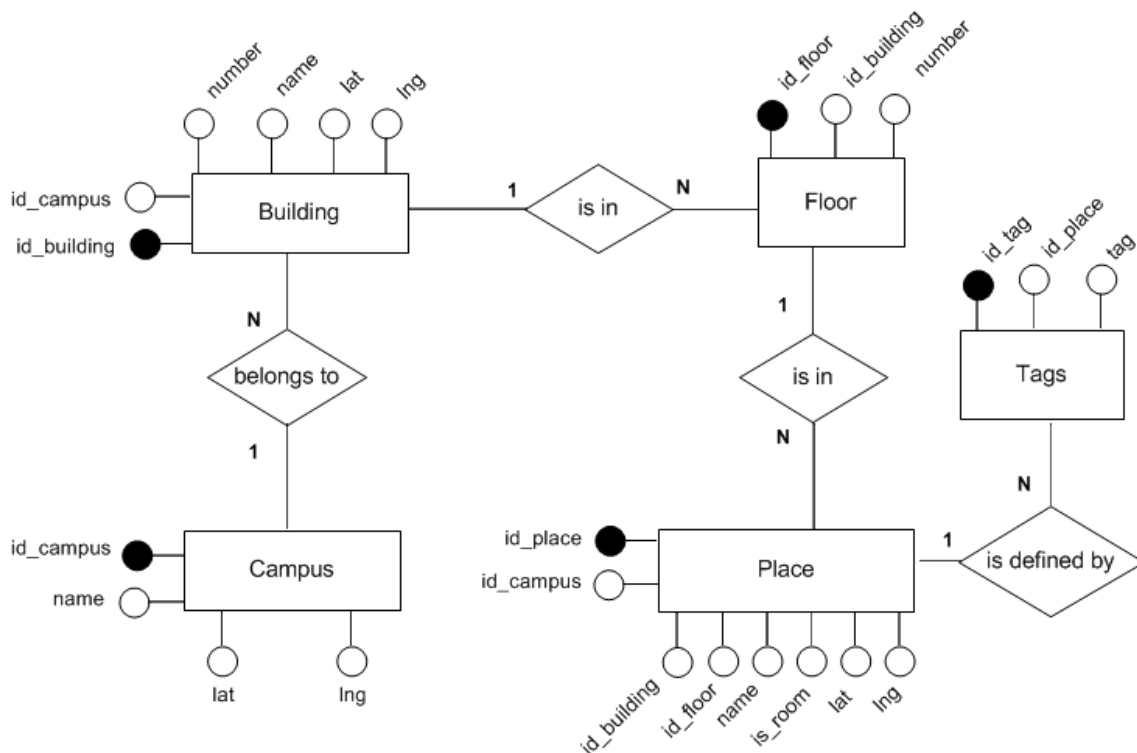


Figura 4.29: Diagrama Entidad Relación de la bases de datos

La transformación al esquema relacional que se ha realizado desde el diagrama Entidad/Relación al modelo relacional para la posterior creación de la base de datos se muestra en la Figura 4.30.

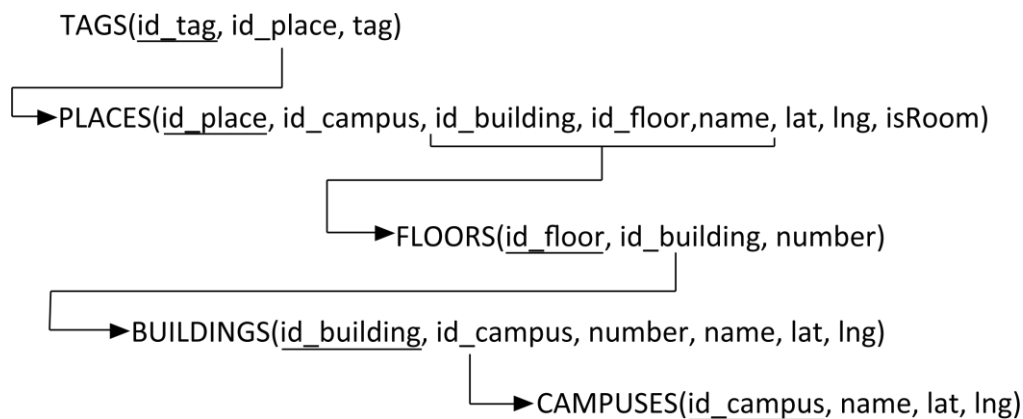


Figura 4.30: Esquema relacional de la base de datos

En líneas generales, en la base de datos se almacena información sobre los campus que componen la UC3M, los edificios que se albergan en los campus, las plantas que tiene cada edificio y los lugares, que pueden ser aulas, despachos o laboratorios, que componen cada planta. Como lugar también se contemplan lugares al aire libre, es decir, que no están dentro de ningún edificio. En la entrada de cada lugar en la base de datos se almacenan las coordenadas geográficas en las que está ubicado.

Además, se almacenan una serie de *tags* o etiquetas asociadas a cada lugar. Los tags son de utilidad para reducir el tiempo de búsqueda. Para un lugar se generan tantas etiquetas como en el campus se conozca al lugar y como el usuario pueda colocar los puntos de los identificadores en el campo de búsqueda de la app. Es común que los usuarios no recuerden fielmente la estructura de los identificadores o que a un lugar le recuerden por su nombre significativo.

La base de datos se diseñó pensando en el identificador que cada lugar de la UC3M tiene asignado. De esta forma, se puede recuperar un lugar de forma más natural a partir del significado de la estructura de los identificadores.

La forma de diseño elegida hace que las peticiones del servicio web tarden más tiempo en ejecutarse ya que tiene que realizar consultas sobre varias tablas pero como ventaja, aporta a la UC3M un repositorio completo de lugares. Para aumentar la velocidad de las peticiones, se decidió incluir la tabla de *tags* ya que el número de consultas se reduce a una consulta.

4.6 DISEÑO DEL REPOSITORIO DE PLANOS

En diferentes secciones de este documento se ha hecho referencia a los planos de las plantas que la app muestra superpuestos sobre el mapa cartográfico. Durante el desarrollo de la aplicación se ha participado en la creación del repositorio que posibilita almacenar y recuperar los planos de las plantas. El repositorio se encuentra incluido en el componente Maps Repository del diagrama de componentes del sistema. En concreto, el autor ha trabajado en el proceso de edición de los planos.

La UC3M facilitó al grupo SEL-UC3M los planos arquitectónicos de las plantas de todos los edificios de la UC3M en soporte digital, imagen vectorial en formato *dxg*.

Los pasos completos a dar hasta llegar al resultado final eran:

1. Editar los planos para eliminar elementos que no tuvieran significado para los usuarios de la app y destacar elementos que pudieran ayudar a los usuarios a desplazarse por el edificio.
2. Redimensionar los planos para que su superposición sobre el mapa cartográfico quede exacta.
3. Crear un archivo por cada planta con la información de su ubicación geográfica.
4. Generar las tiles de cada planta.

El trabajo de edición de los planos se llevó a cabo con una herramienta comercial de edición de imágenes vectoriales. Se eliminaron los elementos sin significado para los usuarios de la app como columnas o representaciones gráficas de mobiliario y se colorearon de verde las puertas que normalmente están abiertas en cada planta y por las que puede pasar cualquier tipo de usuario de la app. También se colorearon de verde los escalones de las escaleras y en rojo la flecha del sentido de subida para facilitar a los usuarios reconocer las escaleras en el plano. Las rampas y su sentido de subida también fueron coloreados.

Desde la herramienta de edición los planos se exportaron a formato *TIFF*. Para cada edificio se realizó una captura de su visualización en el servicio web Google Maps. Debido a que no es posible obtener una vista satélite de los edificios de manera que no se vean inclinados, es necesario abrir todos los archivos *TIFF* de una planta en un programa de edición de imágenes junto con la captura del edificio para redimensionarlos y girarlos de manera que cuando se muestren superpuestos en la app, lo hagan en la proporción y orientación correcta. Se obtendrá un archivo *psd* o similar del edificio con una capa por planta redimensionada y orientada correctamente.

Para dotar de información geográfica a la imagen de cada planta, se exportó la capa de cada planta desde el archivo *psd* a una imagen en formato *TIFF* y se creó un script para cada planta que asociaba un número determinado de píxeles de la imagen a las coordenadas geográficas sobre las que debe de ir superpuesta. El script crea un fichero de tipo *VRT* que abierto con el programa MapTiler, genera la tiles de una planta. Este proceso se explica con detalle en [25].

Por último, se estableció que los directorios del repositorio debían de ordenarse siguiendo la siguiente estructura:

```
http://<URL del servidor>/tiles/<NombreEdificioPlantaNº>/<zoom>/
```

El campo <NombreEdificioPlantaNº> indica que se tiene que crear una carpeta por cada planta de los edificios de la UC3M y en su nombre tiene que aparecer el nombre del edificio y el número de la planta. El campo <zoom> indica el nivel de zoom para el que están generadas las tiles de la planta ya que el programa MapTiler permite generar tiles de una imagen para distintos niveles de zoom. En el caso que nos ocupa, se crearon tiles para los niveles de zoom desde el 17 hasta el 21 incluidos. Es en las carpetas de zoom donde se han almacenado las tiles correspondientes.

La creación del repositorio de planos es una de las tareas que han requerido más esfuerzo ya que se han repetido los pasos que se introducían al principio de esta sección una vez por cada edificio en el que se ha participado en el proceso de edición de sus planos. Aunque es una tarea repetitiva y sencilla de reproducir cuando se conoce el procedimiento, requiere mucha atención para realizar el trabajo con la mayor precisión posible.

CAPÍTULO 5 PRUEBAS DE LA APLICACIÓN

5.1	PRUEBAS UNITARIAS	92
5.1.1	PRIMERA BATERÍA DE PRUEBAS: CONFIGURACIÓN	92
5.1.2	SEGUNDA BATERÍA DE PRUEBAS: BÚSQUEDA DE LUGARES.....	93
5.1.3	TERCERA BATERÍA DE PRUEBAS: BÚSQUEDA DE PERSONAS.....	94
5.1.4	CUARTA BATERÍA DE PRUEBAS: PETICIONES DE APPS EXTERNAS.....	95
5.1.5	QUINTA BATERÍA DE PRUEBAS: DETALLE DE UN LUGAR	96
5.1.6	SEXTA BATERÍA DE PRUEBAS: DETALLE DE UNA PERSONA	97
5.1.7	SÉPTIMA BATERÍA DE PRUEBAS: INFORMACIÓN DE LA APP	98
5.2	PRUEBAS DE INTEGRACIÓN	98
5.3	PRUEBAS DE SISTEMA	99

En este capítulo se definen las pruebas unitarias, de integración y de sistema que permiten verificar el correcto funcionamiento de la aplicación.

Las pruebas de software son una serie de procesos de evaluación del software cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto al cliente.

5.1 PRUEBAS UNITARIAS

Para probar el correcto funcionamiento de la aplicación se han diseñado una serie de pruebas unitarias que prueban la funcionalidad de los componentes que incorporan las funcionalidades de la aplicación. Por tanto, las clases de los componentes para los que se han realizado pruebas unitarias pertenecen al componente Map y al componente Options.

En el proyecto de XCode creado para implementar la aplicación se ha añadido un módulo de pruebas unitarias y se han seguido los pasos para la configuración y creación de pruebas unitarias proporcionados por Apple [26].

El criterio que se ha seguido para definir las pruebas unitarias corresponde a comprobar que la interacción que hace el usuario en la aplicación se vea completada. Las interacciones que realiza el usuario con la aplicación se encuentran definidas en las historias de usuario, sección 3.3, por lo que se comprobará que las secuencias de pasos que componen las historias de usuario puedan ser completadas por los usuarios.

5.1.1 PRIMERA BATERÍA DE PRUEBAS: CONFIGURACIÓN

La primera batería de pruebas se ha diseñado la clase *FlipsideViewController* que es la encargada de manejar la configuración de la app con la finalidad de comprobar que la app almacena correctamente los datos de configuración a nivel interno para poder mostrarlos en vista de configuración.

Se ha probado que la clase almacene, en la estructura interna de datos destinada a almacenar información de configuración, los datos para mostrar el contenido de la vista detalle de configuración y que después los recupere correctamente. Las pruebas que se han realizado han sido:

- Prueba **testConfigurationIsSaved**. Se considera correcta si la estructura de datos encargada de almacenar los datos de configuración almacena en los campos correspondientes las opciones de configuración del usuario.
- Prueba **testConfigurationIsLoaded**. Se considera correcta si la clase recupera correctamente las opciones de configuración seleccionadas por el usuario.

El resultado de la batería de pruebas ha sido satisfactorio, almacenando y recuperando correctamente la clase *FlipsideViewController* los datos de configuración de la app elegidos por el usuario. En la Figura 5.1 se puede ver el resultado de la ejecución de la prueba con datos reales.

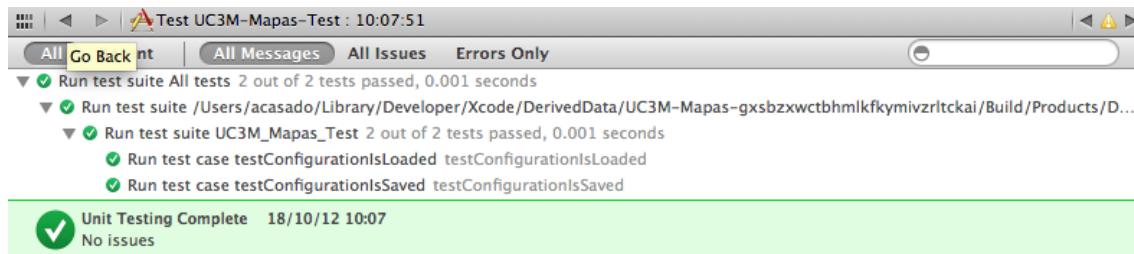


Figura 5.1: Resultado de la batería de pruebas de configuración

5.1.2 SEGUNDA BATERÍA DE PRUEBAS: BÚSQUEDA DE LUGARES

La siguiente batería de pruebas se ha realizado sobre la clase *MainViewController* que es la encargada de la manejar la funcionalidad principal para probar la búsqueda de lugares.

Se ha probado que la app sea capaz de obtener datos del servicio web y que sean los esperados utilizando diferentes términos de búsqueda correspondientes a lugares. Las pruebas que se han realizado han sido:

- Prueba **testSearchPlaceResponseData**. Se considera correcta si para el término de búsqueda utilizado, se obtiene respuesta desde el servidor, es decir, el resultado de la petición no es nulo.
- Prueba **testPlaceExists**. Se considera correcta si los datos que se han obtenido para el término de búsqueda utilizado, se corresponden con datos de un lugar.
- Prueba **testPlaceIsCorrectLocated**. Se considera correcta si las coordenadas del lugar obtenidas se corresponden con las que previamente se conocen del mismo.

Las pruebas de esta batería se han ejecutado para diferentes tipos de términos de búsqueda que son posibles en la búsqueda de lugares. Se han utilizado identificadores estándar como “2.1.C02”, identificadores que no siguen el formato como “21C02”, y nombres por el que se conoce a un lugar en un campus como “cafeteria” o “locomotora”. Para todos los términos el resultado de las pruebas has sido satisfactorio. También se ha efectuado una búsqueda con un término de búsqueda de un lugar que no existe obteniendo una respuesta del servidor que indica que no se han obtenido resultados. La Figura 5.2 muestra el resultado de la segunda batería de pruebas para el término “2.1.C02”.

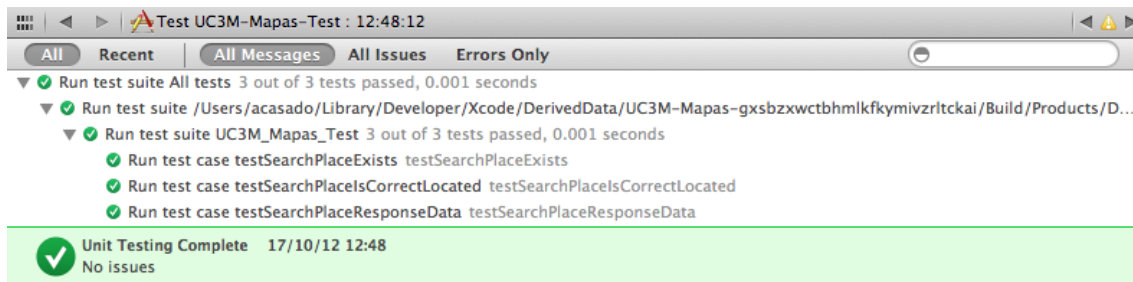


Figura 5.2: Resultado de la batería de pruebas de búsqueda de lugares

5.1.3 TERCERA BATERÍA DE PRUEBAS: BÚSQUEDA DE PERSONAS

La siguiente batería de pruebas se ha realizado sobre la clase *MainViewController* para probar la búsqueda de personas.

Se ha probado que la app sea capaz de obtener datos del servicio web y que sean los esperados utilizando diferentes términos de búsqueda correspondientes a personas. Las pruebas que se han realizado han sido:

- Prueba **testSearchPersonResponseData**. Se considera correcta si para el término de búsqueda utilizado, se obtiene respuesta desde el servidor, es decir, el resultado de la petición no es nulo.
- Prueba **testSearchPersonExists**. Se considera correcta si los datos que se han obtenido para el término de búsqueda utilizado, se corresponden con datos de una persona.
- Prueba **testSearchPersonIsCorrectIdentified**. Se considera correcta si el nombre y los apellidos obtenidos de la persona se corresponden con los que previamente se conocen de la misma.
- Prueba **testSearchPersonIsCorrectLocated**. Se considera correcta si las coordenadas del lugar obtenidas son corresponden con las que previamente se conocen del mismo.

Las pruebas de esta batería se han ejecutado para diferentes tipos de términos de búsqueda que son posibles en la búsqueda de personas. Se han utilizado como términos nombres de personas que existen en el directorio para los que la app obtiene un único resultado como “Garcia Guzman” o para los que obtiene varios como “Garcia”. Para todos los términos el resultado de las pruebas has sido satisfactorio. También se ha efectuado una búsqueda de con un término de búsqueda que no existe obteniendo una respuesta del servidor que indica que no se han obtenido resultados. La Figura 5.3 muestra el resultado de la tercera batería de pruebas para el término “Garcia Guzman”.

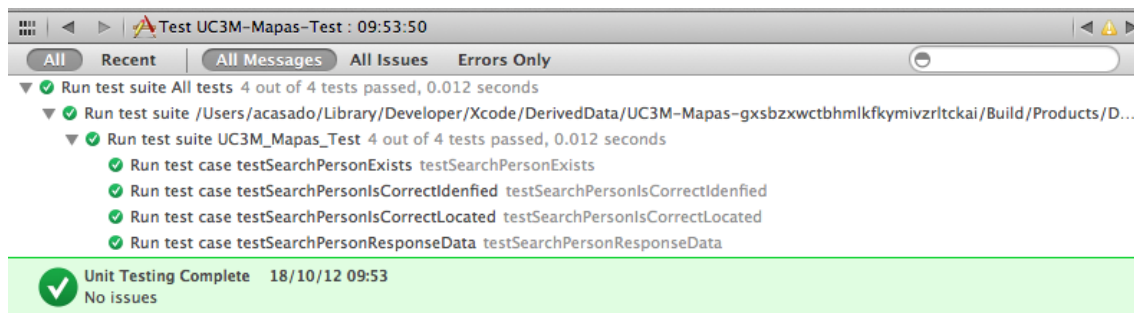


Figura 5.3: Resultado de la batería de pruebas de búsqueda de personas

5.1.4 CUARTA BATERÍA DE PRUEBAS: PETICIONES DE APPS EXTERNAS

Esta batería de pruebas se ha realizado sobre la clase *MainViewController* para probar la recepción de parámetros cuando apps externas solicitan un servicio de búsqueda.

Se ha probado que la app reciba correctamente los parámetros de búsqueda desde otras app y que sea capaz de realizar el tipo de búsqueda correspondiente en función de los parámetros recibidos. Las pruebas que se han realizado han sido:

- Prueba **testExternalParametersReception**. Se considera correcta si los parámetros recibidos se capturan de forma correcta, es decir, que no se tomen valores nulos.
- Prueba **testExternalParametersCorrectSearchDone**. Se considera correcta si la app ha sido capaz de interpretar correctamente los parámetros de la petición. Si la búsqueda pedida es sobre lugares, deberá obtener resultados de búsqueda en la estructura de datos que almacena los resultados de este tipo de búsqueda y en el caso de ser sobre personas, en la estructura de datos correspondiente.

Las pruebas de esta batería se han ejecutado utilizando, para el tipo de búsqueda respectivo, los términos de búsqueda definidos para la segunda y la tercera batería. Cabe recordar que los parámetros que debe de recibir la app para iniciar una búsqueda son el tipo de búsqueda, el campus sobre el que se quiere realizar la búsqueda y el término de búsqueda.

El resultado de la batería de pruebas ha sido satisfactorio. En la Figura 5.4 se muestra el resultado de la batería solicitando una búsqueda de un lugar en el campus de Leganés con el término de búsqueda “2.1.C02”.

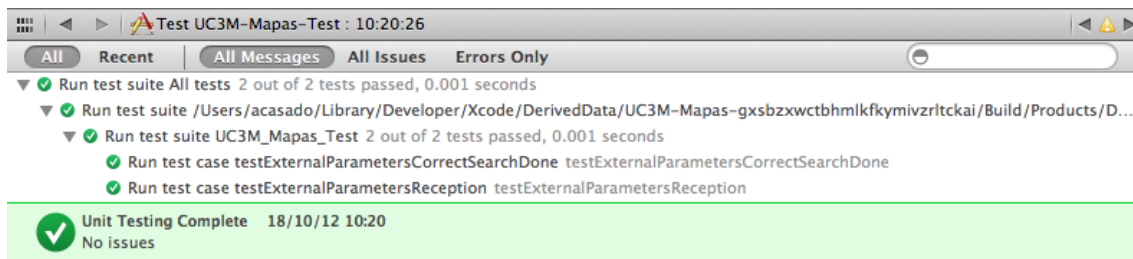


Figura 5.4: Resultado de la batería de pruebas de peticiones externas

5.1.5 QUINTA BATERÍA DE PRUEBAS: DETALLE DE UN LUGAR

La quinta batería de pruebas se ha realizado sobre la clase *PlaceDetailViewController* para comprobar que recibe correctamente los datos, desde la clase *MainViewController*, que la permiten proporcionar a la interfaz de usuario de la vista detalle de lugar la información que tiene que mostrar.

Se ha probado que la clase almacene en la estructura de datos correspondiente los datos para mostrar la información de la vista detalle de lugar. La prueba que se ha realizado ha sido:

- Prueba **testPlaceInfoIsSaved**. Se considera correcta si la estructura de datos encargada de almacenar el los datos del lugar no está vacía.

El resultado de la batería de pruebas ha sido satisfactorio, recibiendo y almacenando correctamente la clase *PlaceDetailViewController* los datos recibidos desde la clase *MainViewController*. En la Figura 5.5 se puede ver el resultado de una ejecución de la batería con datos reales.

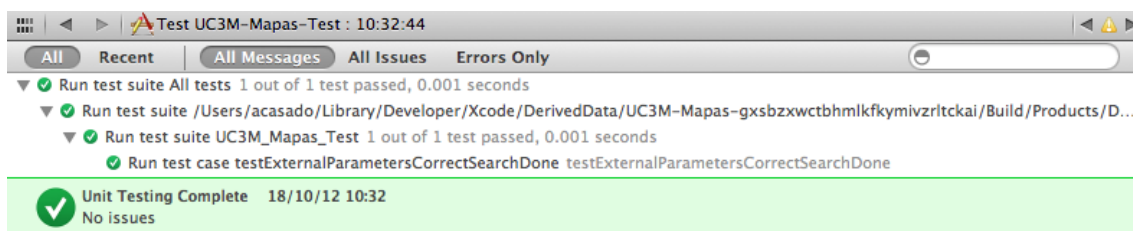


Figura 5.5: Resultado de la batería de pruebas del detalle de un lugar

5.1.6 SEXTA BATERÍA DE PRUEBAS: DETALLE DE UNA PERSONA

La sexta batería de pruebas se ha realizado sobre la clase *PersonDetailViewController* para comprobar que recibe correctamente los datos, desde la clase *MainViewController*, que la permiten proporcionar a la interfaz de usuario de la vista detalle de persona la información que tiene que mostrar.

Se ha probado que la clase almacene en las estructuras de datos correspondientes los datos para mostrar la información de la vista detalle de lugar. Las pruebas que se han realizado han sido:

- Prueba **testNameIsSaved**. Se considera correcta si la estructura de datos encargada de almacenar el nombre completo de la persona no está vacía.
- Prueba **testDepartmentIsSaved**. Se considera correcta si la estructura de datos encargada de almacenar el departamento al que pertenece la persona no está vacía.
- Prueba **testTablePesonInfoIsSaved**. Se considera correcta si la estructura de datos encargada de almacenar los datos de contacto de la persona no está vacía.

El resultado de la batería de pruebas ha sido satisfactorio, recibiendo y almacenando correctamente la clase *PersonDetailViewController* los datos recibidos desde la clase *MainViewController*. En la Figura 5.6 se puede ver el resultado de una ejecución de la batería con datos reales.

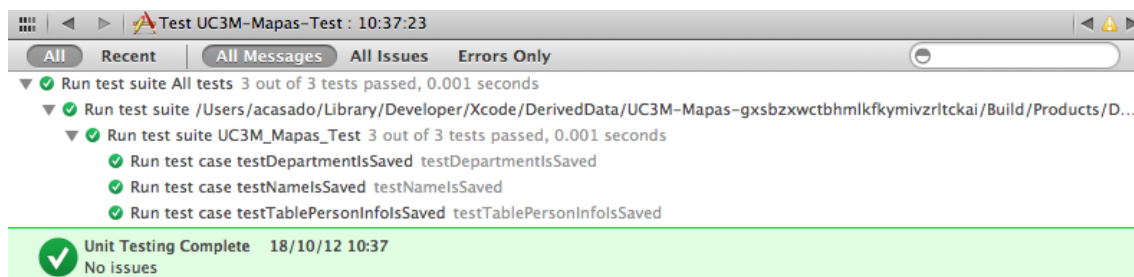


Figura 5.6: Resultado de la batería de pruebas del detalle de una persona

5.1.7 SÉPTIMA BATERÍA DE PRUEBAS: INFORMACIÓN DE LA APP

Esta batería de pruebas se ha realizado sobre la clase *AboutViewController* para comprobar que recupera correctamente los datos que la permiten proporcionar a la interfaz de usuario de la vista acerca de la información que tiene que mostrar.

Se ha probado que la clase recupere en la estructura de datos correspondiente los datos para mostrar el contenido de la vista detalle de acerca de. La prueba que se ha realizado ha sido:

- Prueba **testInfoIsLoaded**. Se considera correcta si la clase recupera correctamente la información de la app.

El resultado de la prueba ha sido satisfactorio, recuperando correctamente la clase *AboutViewController* el contenido de la interfaz de usuario. En la Figura 5.7 se puede ver el resultado de la ejecución de la prueba con datos reales.

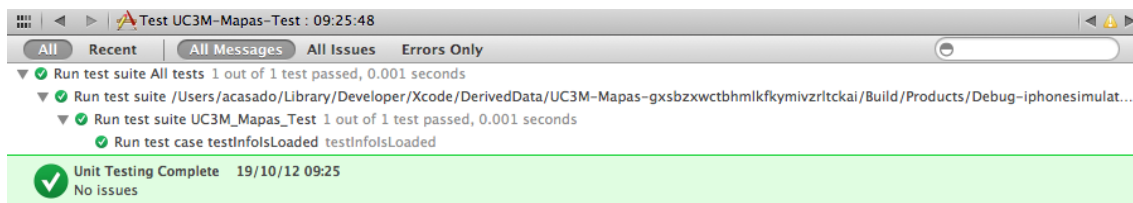


Figura 5.7: Resultado de la batería de pruebas de la información de la app

5.2 PRUEBAS DE INTEGRACIÓN

La aplicación UC3M Mapas se comunica externamente con un servidor, el del SEL-UC3M, al que le realiza las peticiones de consulta de las búsquedas de lugares y de personas e internamente, con *frameworks* del sistema operativo y librerías desarrolladas por terceros que dan soporte a las funcionalidades de la app.

Desde el comienzo del desarrollo de la app se fue realizando la integración entre los distintos componentes por lo que no se ha considerado realizar pruebas de integración al finalizar el desarrollo.

5.3 PRUEBAS DE SISTEMA

En línea con el tipo de metodología utilizada para el desarrollo del proyecto, SCRUM, en las pruebas de sistema se ha seguido la misma filosofía. En cada iteración se han realizado pruebas de sistema para que el cliente, en este caso el tutor del proyecto, valorase si el proyecto avanzaba como se deseaba. Como clave en la definición y ejecución de estas pruebas se deben de considerar los siguientes puntos:

- El cliente debe tomar parte activa en la creación de las pruebas de sistema.
- En muchas ocasiones no se realizan todas las pruebas de sistema hasta que no se entrega el software al cliente. Las pruebas de aceptación se deben proporcionar en cada iteración siendo deseable que estén listas a la mitad de la misma.
- Únicamente se deben probar las cosas que quiere que funcionen.
- Es imprescindible especificar pruebas de sistema para cada diagrama de secuencia definido en el diseño de la aplicación.

Cada una de las pruebas deberá de ser definida por los siguientes campos:

- **ID:** el identificador de la prueba compuesto por la cadena PS- y dos valores separados por un punto. El primero de ellos identifica el número de prueba. El segundo identifica el diagrama de secuencia al que se refiere.
- **Descripción:** es el objetivo que tiene marcado la prueba para que sea pasada.
- **Pasos:** serie de pasos que se ha de hacer en la aplicación para poder realizar la prueba.
- **Errores posibles:** se enumeran los posibles errores que se han encontrado o pueden encontrarse al realizar la prueba, y que no permiten que se considere pasada.
- **Requisitos:** se enumeran los requisitos con los que está relacionada la prueba.
- **Resultado:** breve explicación del resultado obtenido.
- **Estado:** tras realizar la prueba es el resultado de la misma. Será “pasada” si es positivo y “no pasada” si es negativo.

En la Tabla 5.1 se muestra la plantilla utilizada para definir las pruebas de sistema.

ID
Descripción
Pasos
Errores posibles
Requisitos
Resultado
Estado

Tabla 5.1: Plantilla de prueba de sistema

La primera prueba de sistema, especificada en la Tabla 5.2 prueba que el inicio de la aplicación sea correcto, es decir: la aplicación deberá iniciarse sin que se muestre ninguna advertencia acerca de la conexión de red o de la conexión con el servicio web.

ID	PS-1.1
Descripción	Se realiza la inicialización de la aplicación.
Pasos	1. Pulsar el icono de la app UC3M Mapas en el terminal.
Errores posibles	<ul style="list-style-type: none"> • No es posible establecer conexión con Internet. • La vista principal no carga.
Requisitos	RI-003
Resultado	Se inicia la app UC3M Mapas en la vista principal.
Estado	PASADA

Tabla 5.2: PS-1.1 - Inicio de la aplicación

En la siguiente prueba definida, ver Tabla 5.3, se comprueba la visualización de los diferentes tipos de mapas cartográficos en la vista principal seleccionándolos en la configuración de la aplicación. Los tipos de mapa disponibles son normal, satélite e híbrido.

ID	PS-2.2
Descripción	Comprobar que se muestran correctamente los diferentes tipos de mapa.
Pasos	<ol style="list-style-type: none">1. Iniciar la aplicación.2. Acceder a la vista de configuración.3. Elegir el tipo de mapa normal.4. Volver a la vista principal.5. Acceder a la vista de configuración.6. Elegir el tipo de mapa satélite.7. Volver a la vista principal.8. Acceder a la vista de configuración.9. Elegir el tipo de mapa híbrido.10. Volver a la vista principal.
Errores posibles	<ul style="list-style-type: none">• No es posible establecer conexión con Internet.• No es posible establecer conexión con el servidor del mapa o no puede completar la acción requerida.• La vista de configuración no carga.• La vista principal no carga.
Requisitos	RF-011, RI-003, RI-004
Resultado	Tras acceder a la vista de configuración desde la vista principal y seleccionar cada tipo de mapa, se comprueba en la vista principal que se carga el tipo de mapa seleccionado.
Estado	PASADA

Tabla 5.3: PS-2.2 - Comprobación de tipos de mapas

La siguiente prueba consiste en comprobar que la aplicación es capaz de encontrar un lugar buscado por un usuario. En la Tabla 5.4 se muestra el procedimiento realizado y el resultado para el término de búsqueda utilizado, “2.1.C02”.

ID	PS-3.3
Descripción	Realizar la búsqueda de un lugar usando el término de búsqueda “2.1.C02”.
Pasos	<ol style="list-style-type: none"> 1. Seleccionar en la vista de configuración el campus en el que se desea realizar la búsqueda en caso de que se desee uno diferente al actualmente elegido. 2. Seleccionar en la vista de configuración el tipo de mapa en caso de que se desee uno diferente al actualmente elegido. 3. Seleccionar en la vista principal el tipo de búsqueda de lugar en caso de que no esté seleccionada. 4. Introducir en la vista principal el término de búsqueda. 5. Pulsar el botón de búsqueda.
Errores posibles	<ul style="list-style-type: none"> • No es posible establecer conexión con Internet. • No es posible establecer conexión con el servicio web del SEL-UC3M. • El servicio web del SEL-UC3M no puede completar la acción solicitada. • El lugar buscado no está georreferenciado en la base de datos.
Requisitos	RF-001, RF-002, RF-010, RF-011, RI-003, RI-010, RR-001
Resultado	La app superpone en el mapa el plano de la planta del edificio en el que se encuentra el lugar y sobre el plano indica la ubicación del lugar. Además informa del campus y del plano que se están visualizando.
Estado	PASADA

Tabla 5.4: PS-3.3 - Prueba de sistema de búsqueda de un lugar

Seguidamente, tras pasar con éxito la prueba 3.3, se ha probado la funcionalidad que permite reportar una incidencia al administrador. En la Tabla 5.5 se muestra el proceso llevado a cabo para reportar una incidencia partiendo de una búsqueda previa de lugar y el resultado obtenido.

ID	PS-4.3
Descripción	Reportar una incidencia de un lugar previamente buscado.
Pasos	<ol style="list-style-type: none">1. Pulsar el botón del globo de texto situado en la ubicación del lugar previamente buscado para acceder a la vista de detalle del lugar.2. Pulsar el botón de incidencias, situado en la parte derecha de la barra superior de la vista de detalle de lugar.3. Rellenar el correo electrónico siguiendo el esquema recomendado y enviarlo a la dirección indicada.4. Volver a la aplicación UC3M Mapas.
Errores posibles	<ul style="list-style-type: none">• No es posible establecer conexión con Internet.• La aplicación de correo no está instalada o no se puede iniciar.
Requisitos	RF-003, RF-005, RI-003, RI-005
Resultado	Tras pulsar el botón del globo de texto, la app carga la vista de detalle del lugar y una vez pulsado el botón de incidencia, abre la aplicación de correo con una plantilla preparada para enviar. Se ha podido escribir y enviar el correo y volver a la aplicación.
Estado	PASADA

Tabla 5.5: PS-4.3 - Prueba de sistema de reporte de incidencia de un lugar

La siguiente prueba consiste realizar una búsqueda de persona en el campus de Getafe. El término de búsqueda de persona utilizado ha sido “daniel peña”. Como resultado, la app deberá ubicar sobre un plano el lugar donde se encuentra el actual rector de la UC3M en el campus de Getafe.

En la Tabla 5.6 se muestra tanto el procedimiento seguido como el resultado obtenido tras la realización de la prueba.

ID	PS-5.4
Descripción	Realizar la búsqueda de una persona usando el término de búsqueda “daniel peña”.
Pasos	<ol style="list-style-type: none"> 1. Seleccionar en la vista de configuración el campus en el que se desea realizar la búsqueda en caso de que se desee uno diferente al actualmente elegido. 2. Seleccionar en la vista de configuración el tipo de mapa en caso de que se desee uno diferente al actualmente elegido. 3. Introducir en la vista principal el término de búsqueda. 4. Pulsar el botón de búsqueda.
Errores posibles	<ul style="list-style-type: none"> • No es posible establecer conexión con Internet. • No es posible establecer conexión con el servicio web del SEL-UC3M. • El servicio web del SEL-UC3M no puede completar la acción solicitada. • La persona buscada no se encuentra en el campus seleccionado. • El despacho obtenido de la persona buscada no está georreferenciado en la base de datos.
Requisitos	RF-001, RF-002, RF-010, RF-011, RI-003, RI-010, RI-001
Resultado	La app superpone en el mapa el plano de la planta del edificio en el que se encuentra el lugar de la persona buscada y sobre el plano indica la ubicación del lugar. Además informa del campus y del plano que se están visualizando.
Estado	PASADA

Tabla 5.6: PS-5.4 - Prueba de sistema de búsqueda de una persona

La siguiente prueba, definida en la Tabla 5.7, consiste en probar la funcionalidad de todas las acciones disponibles desde la vista de detalle de persona que implican a sus datos de contacto. Las posibles acciones son enviarla un correo electrónico, llamarla por teléfono y agregarla a la agenda de contactos del dispositivo. La prueba se ha realizado partiendo del resultado de la prueba P5.4.

ID	PS-6.4
Descripción	Ejecutar las acciones de la vista de detalle de una persona, previamente buscada, que impliquen a los datos de contacto.
Pasos	<ol style="list-style-type: none"> 1. Pulsar el botón del globo de texto situado en la ubicación del lugar previamente buscado para acceder a la vista de detalle del lugar. 2. Pulsar sobre el correo electrónico que muestra la vista de detalle de la persona. 3. Redactar el correo y enviarlo a la dirección indicada. 4. Volver a la vista de detalle de la persona. 5. Pulsar sobre el número de teléfono mostrado que muestra la vista de detalle de la persona. 6. Llamar al número indicado. 7. Volver a la vista de detalle de persona. 8. Pulsar el botón de la parte superior de la vista para ver las opciones posibles. 9. Seleccionar la opción de agregar a contactos. 10. Guardar el contacto. 11. Volver a la vista de detalle de persona.
Errores posibles	<ul style="list-style-type: none"> • No es posible establecer conexión con Internet. • La aplicación de correo no está instalada o no se puede iniciar. • La aplicación de teléfono no está instalada o no se puede iniciar. • La aplicación de contactos no está instalada o no se puede iniciar.
Requisitos	RF-004, RF-006, RF-007, RF-008, RI-003, RI-006
Resultado	Tras pulsar el botón del globo de texto, la app carga la vista de detalle de la persona. Se han podido completar correctamente todas las acciones de la vista de detalle de persona que implican a los datos de contacto.
Estado	PASADA

Tabla 5.7: PS-6.4 - Prueba de sistema de la vista de detalle de persona

Finalmente se realiza una prueba para comprobar las acciones que se pueden realizar desde la vista de acerca de que ofrece la aplicación y al finalizar, se comprueba el cierre de la aplicación. La prueba se ha definido en la Tabla 5.8.

ID	PS-7.5
Descripción	Comprobar las acciones disponibles desde la vista acerca de.
Pasos	<ol style="list-style-type: none"> 1. Desde la vista principal, pulsar el botón de la esquina superior izquierda de la vista principal correspondiente a créditos y ayuda. 2. Pulsar sobre el texto “Ayuda”. 3. Visualizar las indicaciones de ayuda. 4. Cerrar las indicaciones de ayuda. 5. Pulsar sobre el texto “Contáctanos”. 6. Escribir un correo electrónico a la dirección de soporte. 7. Volver a la vista acerca de. 8. Pulsar sobre el botón del texto “Twitter: @seluc3m” 9. Navegar en Twitter. 10. Volver a la vista de acerca de. 11. Pulsar sobre el botón del texto “Créditos”. 12. Visualizar la vista de créditos. 13. Volver a la vista acerca de. 14. Cerrar la aplicación.
Errores posibles	<ul style="list-style-type: none"> • No es posible establecer conexión con Internet. • La vista de acerca de no carga. • La vista de ayuda no carga. • La vista de créditos no carga • La aplicación de correo no está instalada o no se puede iniciar. • No es posible establecer conexión con Internet.
Requisitos	RF-009, RI-003, RI-007, RI-008, RI-009
Resultado	Se han completado todas las acciones disponibles en la vista acerca de y se ha cerrado la aplicación correctamente.
Estado	PASADA

Tabla 5.8: PS-7.5 Prueba de sistema de la vista de acerca de

Todas las pruebas de sistema han sido pasadas correctamente pudiendo de esta manera distribuir la aplicación a los usuarios a través de la App Store. Antes de realizar el lanzamiento oficial y después de que los programadores hicieron pruebas de manera informal a lo largo de las iteraciones, la aplicación pasó una fase de pruebas en la que diferentes usuarios de la comunidad universitaria utilizaron la app y reportaron los fallos que encontraron. En esa fase los usuarios no reportaron fallos que impidieran la utilización de la app y se aprovechó para mejorar el rendimiento de la app.

CAPÍTULO 6 CONCLUSIONES Y LÍNEAS FUTURAS

6.1	CONCLUSIONES.....	110
6.2	LÍNEAS FUTURAS.....	112
6.2.1	MEJORA EN LA LOCALIZACIÓN EN INTERIORES.....	112
6.2.2	RUTAS ENTRE LUGARES.....	113
6.2.3	GEOLOCALIZACIÓN DE OBJETOS.....	114
6.2.4	AMPLIACIÓN DE INFORMACIÓN EN EL DETALLE DE UN LUGAR.....	114
6.2.5	RED SOCIAL PARA LOCALIZACIÓN DE AMIGOS.....	114
6.2.6	MANTENIMIENTO DE LA BASE DE DATOS Y MEJORA DE PLANOS.....	115

En este capítulo se incluyen las conclusiones obtenidas durante el desarrollo del proyecto las cuales tratan sobre el entorno del proyecto y de las experiencias del autor en su elaboración y también las líneas futuras en las que se tratan nuevas funcionalidades que es posible incorporar a la aplicación para su mejora.

6.1 CONCLUSIONES

Hoy en día los seres humanos sienten la necesidad de conocer información acerca del entorno que les rodea y, en menor medida, de otros entornos externos, con la finalidad de procesarla para convertirla en conocimiento que revierta principalmente en la mejora de la vida diaria. En cada entorno, la mejora de la vida diaria puede traducirse de diferentes maneras en función del ámbito de aplicación. Así, en un ámbito empresarial, podrá significar una ventaja competitiva frente a otra entidad o, en un ámbito sanitario, la mejora de la calidad de asistencia a los pacientes. El único punto en común en todos los ámbitos es la supervivencia del individuo o el colectivo en el entorno.

El proyecto fin de carrera realizado pertenece a una línea de investigación recientemente abierta en la Universidad Carlos III de Madrid: desarrollo de apps para la mejora de la vida universitaria. Ha sido promovida por el grupo de investigación Software Engineering Lab del Departamento de Informática y cuenta con la colaboración del Servicio de Informática y Comunicaciones, el Servicio de Comunicación Institucional, del Vicerrectorado de Infraestructuras y Medio Ambiente y del Vicerrectorado de Cultura y Comunicación. La cooperación de las personas involucradas de cada grupo ha sido fundamental para poner a disposición de la comunidad universitaria de la UC3M la primera versión de un ecosistema de aplicaciones orientadas a mejorar y facilitar su vida diaria en la universidad.

La aplicación desarrollada, UC3M Mapas, se enmarca dentro de la línea de investigación de mejora de la vida universitaria. Se pretende con ella cubrir las necesidades recurrentes de los miembros de la comunidad universitaria en cuanto a la localización de lugares, como despachos o aulas. Esta es una necesidad común en todos los entornos cuando no se conoce el espacio físico al que pertenecen. Gracias al trabajo desarrollado para elaborar una base de datos que almacene la correspondencia entre un lugar de la UC3M y su coordenada geográfica, unido al trabajo realizado para adaptar los planos de las plantas de los edificios de la Universidad, se ha dado un gran primer paso para proveer servicios de localización en interiores.

Dado que la UC3M dispone de la relación entre un trabajador y su despacho, se ha incluido a la localización en interiores de lugares, la localización en interiores de personas. Con respecto a la localización *indoor* de lugares, ha sido necesario utilizar un servicio que provee la búsqueda de personas dando información sobre su despacho. Conocido el identificador del despacho, ha sido bastante sencillo usar la infraestructura desplegada para ubicar el lugar donde se encuentra el despacho en la UC3M.

Las posibilidades de la tecnología utilizada han sido bien aprovechadas ya que el desarrollo de la aplicación no se ha limitado a ubicar un lugar buscado por el usuario, sino que es posible acceder a más información relacionada con el lugar. En el caso de la búsqueda de personas, es posible interactuar con la información del lugar para ponerse en contacto con la persona. Este hecho aporta un valor añadido a la aplicación.

Con el conocimiento que ofrece la aplicación a los usuarios con la combinación de la información del punto geográfico de un lugar y los planos adaptados, estos pueden acceder de una manera rápida a la ubicación de los lugares y a orientarse en los edificios facilitándoles la navegación en los edificios. Con la provisión de la búsqueda de lugares y de personas en una misma aplicación se produce la unificación de dos servicios que ofrecen una ventaja con la que antes no se contaba, el conocimiento del lugar exacto donde se encuentra una persona en la universidad y cómo otro miembro de la comunidad universitaria puede llegar hasta él haciendo uso de la información de los planos.

Las dificultades para llevar a cabo el proyecto que se han encontrado se pueden clasificar en las inherentes a la tecnología utilizada y las propias del proceso de desarrollo. Sobre el primer tipo, la primera barrera a romper ha sido aprender a utilizar el entorno de desarrollo, XCode y aprender y entender el lenguaje de programación, Objective-C, y la filosofía de desarrollo para iOS. El entorno XCode es bastante intuitivo de utilizar, encontrándose los mayores problemas a la hora de agregar y configurar los certificados de desarrollo y distribución necesarios para programar, probar y mandar a la AppStore la app.

La filosofía de desarrollo de Apple se basa en desarrollar aplicaciones sencillas basadas en la guía de desarrollo que proporciona a los desarrolladores. Es importante cumplir las pautas principales porque es posible que Apple no apruebe una aplicación como fue el caso de la app contenedora UC3M al parecerse al escritorio iOS para iPhone y iPod Touch. También hay que cuidar el diseño interno de la aplicación y procurar utilizar en la medida de lo posible, como en la mayoría de desarrollos de aplicaciones, patrones de diseño. El desarrollo de aplicaciones iOS está pensado para diseñarlas utilizando el patrón de diseño Modelo Vista Controlador. Es un patrón actualmente bastante extendido y se ajusta bien al desarrollo de aplicaciones móviles aunque la forma de aplicarlo en iOS, unido a características de diseño del lenguaje Objective-C, levanta algunas dificultades que es necesario sortear para apps en las que a su diseño no es aplicable el patrón Modelo Vista Controlador. En la línea del uso del patrón mencionado, al conocer el autor únicamente lenguajes de programación orientados a objetos que hacen uso de un método principal encargado de gestionar la aplicación, éste tuvo que comprender a programar sin hacer uso de un método principal puesto que el control de la ejecución en las apps iOS, se realiza en la clase que maneja la vista de usuario al reaccionar a eventos del mismo sin necesidad de instanciar implícitamente la clase como se haría en un método principal.

Sobre las dificultades propias del proceso de desarrollo cabe destacar dos. La primera es acerca de la nomenclatura de los identificadores de los lugares. Existen identificadores para los que la nomenclatura se ve alterada porque incluyen cadenas de caracteres no esperadas, como *bis*, para referirse a uno de los lugares resultantes de la división de despachos o laboratorios. Por otro lado, la nomenclatura no es homogénea en todos los campus de la UC3M debido a la disposición interior de los edificios. El

interior de los edificios del campus de Getafe carece de zonas identificables por lo que en el diseño de la base de datos se decidió que la parte mínima identificable en un campus fuera la letra de la zona de la planta de un edificio, en caso de que tuviera, mas el número de aula, despacho o laboratorio de la zona o edificio de esos lugares.

La segunda dificultad propia del proceso de desarrollo tiene que ver con la importancia que el personal de la universidad da a mantener sus datos actualizados en el directorio. Es una dificultad difícil de controlar por el autor pero que influye negativamente en el éxito de la aplicación. Por el momento son pocos los casos detectados durante las pruebas realizadas a la aplicación en la que el identificador del despacho no coincide con que actualmente ocupa una persona. También se han detectado situaciones en las que los planos de las plantas de los edificios no estaban actualizados por la redistribución de los espacios físicos. Es deber del personal mantener sus datos actualizados en el directorio LDAP de la UC3M y de notificar y actualizar los cambios que se produzcan en los espacios de la Universidad.

La experiencia del autor durante el desarrollo del proyecto ha sido muy positiva. Ha tenido la posibilidad de elaborar el proyecto en un ambiente de trabajo real y la ayuda de los compañeros del equipo al que pasó a integrar ha sido sin duda gratificante. Sumar a esta experiencia la motivación que siempre produce el aprendizaje de nuevas tecnologías y conocimientos y cómo es de aplicación en necesidades reales de los seres humanos. Además, se ve intensificada por el espíritu que persigue la línea de investigación en la que se enmarca, las actividades docentes y de investigación a las que se dedica el grupo SEL y el uso de tecnologías de reciente aparición utilizadas para dar solución al problema planteado. Las tecnologías móviles han vivido un auge y expansión muy importantes desde los últimos cuatro años y están muy cerca de cambiar la forma en la que se procesa la información y marcar un hito en la historia de la era del conocimiento.

6.2 LÍNEAS FUTURAS

Tras desarrollar las conclusiones obtenidas por el autor del proceso del desarrollo del proyecto fin de carrera, se proponen líneas futuras sobre las que trabajar para dotar de mayor funcionalidad a la app desarrollada.

6.2.1 MEJORA EN LA LOCALIZACIÓN EN INTERIORES

La mejora en la localización en interiores, o localización *indoor*, es un problema sobre el que actualmente se está trabajando en diferentes entidades.

La tecnología de localización más precisa con la que se cuenta es sistema de satélites que proveen el servicio de GPS. El error cometido en indicar la posición de una persona u objeto es bastante pequeño si se encuentra fuera de un edificio, pero suficientemente grande si se encuentra dentro de uno, llegando incluso a situarlo a una persona en el exterior en las pruebas que se han realizado en la aplicación.

En la aplicación, la localización del usuario se obtiene desde varias fuentes: la señal del teléfono, la señal de los routers wifi geolocalizados por el servicio de cartografía o, en última instancia, por los satélites GPS. Resulta necesario que el usuario conozca con un buen nivel de precisión dónde está para poder hacer uso de otras funcionalidades que es posible incorporar a la app.

Existen diferentes alternativas que pueden implementarse para que el usuario conozca su posición:

- Georreferenciar los routers de la red wifi de la UC3M: tiene como inconveniente la distorsión de la señal wifi cuando atraviesa paredes. El objetivo sería triangular las señales wifi más cercanas al usuario para obtener un punto.
- Usar dispositivos RFID: se puede considerar desplegar balizas que contengan chips RFID's en lugares estratégicos del campus, como las entradas a los edificios y los accesos a los pasillos. Ya existen dispositivos móviles que incorporan lectores RFID, por lo que cada vez que el usuario pase por uno, la aplicación le avisaría de dónde se encuentra. La desventaja de esta alternativa es la inversión a realizar en el despliegue de las balizas.
- Usar códigos QR: se basa en el mismo principio que el uso de RFID's con la desventaja de que la localización no se realiza de forma automática. Si sería posible ofrecer una localización precisa porque hacer códigos QR resulta barato en comparación con las balizas RFID y se podrían colocar en cada uno de los lugares de la UC3M. El usuario leería con la app un código QR y esta le muestra en el plano correspondiente con su ubicación.

Conviene valorar las ventajas y desventajas de cada una de las alternativas y decirse por una en función de las necesidades de los usuarios y el presupuesto disponible para ejecutarla.

6.2.2 RUTAS ENTRE LUGARES

Una necesidad latente y que se puede cubrir tomando como punto de partida UC3M Mapas es realizar la ruta entre dos lugares de la UC3M. Resulta muy importante conocer con la mayor precisión la posición del usuario para indicarle el camino desde su ubicación hasta un lugar. También se podría habilitar la construcción de una ruta entre dos lugares.

El mecanismo para implementar la funcionalidad podría consistir en la utilización de los algoritmos de búsqueda de caminos que se pueden encontrar en la literatura de la Teoría de Grafos. En una primera versión, se tomarían como nodos los lugares, incluyéndose en la base de datos algunos destinados a este propósito en puntos clave de los campus, y se generaría automáticamente el camino en función de la prioridad del usuario en cuando a distancia o número de nodos del camino. En versiones posteriores se podrían incluir alternativas de caminos teniendo en cuenta aspectos ambientales como la meteorología o relacionados con discapacidades.

6.2.3 GEOLOCALIZACIÓN DE OBJETOS

En la UC3M se cuenta con material inventariado que da el soporte logístico necesario para el desempeño de las funciones de los miembros de la comunidad universitaria.

Con la geolocalización de objetos se consigue ofrecer nuevos servicios. Se podrían georreferenciar las estanterías de las bibliotecas de la Universidad para indicar a los usuarios dónde encontrar exactamente un libro y no tener que recorrer toda la sala o siguiendo el mismo principio, georreferenciar los armarios de las taquillas de los campus para indicar a los usuarios de éstas donde se encuentra exactamente una taquilla que están visualizando en la aplicación de reserva. Si también se georreferencian los proyectores de las aulas docentes, se puede dar soporte a servicios que mejoren la eficiencia del proceso de reparación de los mismos. Como último ejemplo, georreferenciar los desfibriladores de los campus puede ser muy útil para reducir el tiempo de actuación para salvar la vida de una persona por una persona que no sabe dónde están ubicados.

6.2.4 AMPLIACIÓN DE INFORMACIÓN EN EL DETALLE DE UN LUGAR

Dado que la UC3M dispone de información de ocupación diaria de las aulas, de las prestaciones que cada aula ofrece y del número de ordenadores que se están usando en un determinado momento en un aula informática, se podría aprovechar esta información para mostrar a los usuarios de la app el horario de las aulas con el fin de que puedan hacer reservas de las mismas directamente desde la app conociendo previamente su equipación o en caso de las aulas informáticas, indicar el número de ordenadores en uso con respecto al total para saber si se tiene que dirigir a ese aula en busca de un ordenador libre.

6.2.5 RED SOCIAL PARA LOCALIZACIÓN DE AMIGOS

La búsqueda de personas implementada en la aplicación permite proveer la localización de los despachos de un trabajador de la universidad, siempre y cuando conste en el directorio LDAP.

Esta funcionalidad se podría mejorar ofreciendo la ubicación en tiempo real de los miembros de la comunidad universitaria que hayan permitido al usuario conocer su ubicación. No se pretende saber la localización exacta, pero si por ejemplo el edificio en el que se encuentra la persona para poder saber dónde encontrarla, en caso de que esté en la UC3M, en situación de necesidad.

6.2.6 MANTENIMIENTO DE LA BASE DE DATOS Y MEJORA DE PLANOS

La UC3M está en proceso de expansión y está construyendo nuevos edificios en los campus de Leganés y Getafe y acaba de firmar un convenio de uso del mercado Puerta de Toledo de Madrid con fines educativos. Para los nuevos edificios, como tarea de mantenimiento, es necesario incorporar a la base de datos la georreferenciación de sus lugares y adaptar los planos arquitectónicos para su visualización en la app. Dentro del mantenimiento de la base de datos se encuentra actualizar ubicaciones erróneas de lugares o incluir la georreferenciación de nuevos por motivos de redistribución del espacio físico de la UC3M.

Como mejoras a los planos, se pueden diseñar para que se el usuario se oriente fácilmente, adaptarlos para usuario con algún tipo de discapacidad o añadir indicadores representativos de elementos comunes como aseos, escaleras y ascensores para una mejor interpretación de los mismos. Una opción a tener en cuenta es realizar planos esquemáticos dibujados de las plantas que representen de una forma limpia y fiel los lugares y los elementos que la componen.

BIBLIOGRAFÍA

- [1] ALONSO, Juan Carlos; GARCÍA, Javier; AMESCUA, Antonio. “The use of mobile applications in Spanish Universities, current status and future challenges”. Disponible en: Actas de la International Conference of Education, Research and Innovation (ICERI), (Madrid, España 14-16 de noviembre de 2011).
- [2] Google Inc. “Maps”. [Aplicación Android], 2012. Disponible en Google Play: <https://play.google.com/store/apps/details?id=com.google.android.apps.maps>
- [3] McClendon, B. “A new frontier for Google Maps: mapping the indoors”. [Documento online, Official Google Blog], Noviembre 2011. Disponible en: <http://googleblog.blogspot.com.es/2011/11/new-frontier-for-google-maps-mapping.html>
- [4] Point Inside. “Point Inside Shopping & Travel”. [Aplicación iOS], 2011. Disponible en App Store: <https://itunes.apple.com/us/app/point-inside-shopping-travel/id338171893?mt=8>
- [5] Korea Trade Network. “myCoex”. [Aplicación iOS]. Disponible en App Store: <https://itunes.apple.com/us/app/my-coex/id395472288?mt=8>
- [6] Synergia Media Corp. “Mall Rat”. [Aplicación iOS], 2011. Disponible en App Store: <https://itunes.apple.com/ca/app/mall-rat/id489579704?mt=8>
- [7] Wifarer Inc. “Wifarer”. [Aplicación iOS], 2011. Disponible en App Store: <https://itunes.apple.com/us/app/wifarer/id445140849?mt=8>
- [8] Universidad de Granada. “Universidad de Granada”. [Aplicación iOS]. Disponible en App Store: <https://itunes.apple.com/es/app/universidad-de-granada/id436735938?mt=8>
- [9] Blackboard Inc. “University of Alabama”. [Aplicación iOS]. Disponible en App Store: <https://itunes.apple.com/us/app/university-of-alabama/id432956004?mt=8>
- [10] Massachusetts Institute of Technology. “MIT Mobile”. [Aplicación iOS], 2012. Disponible en App Store: <https://itunes.apple.com/us/app/mit-mobile/id353590319?mt=8>
- [11] Comunidad Movistar. “Las aplicaciones más descargadas en 2012”. [Documento online], 31/03/2012. Disponible en: <http://comunidad.movistar.es/t5/Blog-Smartphones/Las-aplicaciones-m%C3%A1s-descargadas-en-2012/ba-p/495995>
- [12] THE APP DATE. “Informe Apps primer trimestre 2012”. [Documento online]. Disponible en: <http://madrid.theappdate.com/informe-apps/>>
- [13] Solé, O. “Análisis de aplicaciones descargadas: Android supera a iPhone en España”. [Documento online], 21/12/2011. Disponible en: <http://www.celularis.com/software/analisis-de-aplicaciones-descargadas-android-supera-a-iphone-en-espana/>

- [14] Wikipedia. “iOS version history”. [Documento online], 24/10/2012. Disponible en: http://en.wikipedia.org/wiki/IOS_version_history
- [15] Ray, J. “iPhone Application Development in 24 hours” Second Edition. [Documento Online], 15/10/2011. Disponible en: <http://proquest.safaribooksonline.com/book/programming/mobile/9780132601214>
- [16] Sadun, E. “The iOS 5 Developer’s Cookbook: Core Concepts and Essential Recipes for iOS Programmers”, Third Edition. Addison-Wesley Professional. 14/11/2011. [Documento Online], Disponible en: <http://proquest.safaribooksonline.com/book/programming/mobile/9780132613477>
- [17] Apple Developer. “Developer Tools”. [Documento online]. Disponible en: <https://developer.apple.com/technologies/tools/>
- [18] Apple Developer. “The Objective-C Programming Language”. [Documento online], Última actualización 12/10/2011. Disponible en: <https://developer.apple.com/library/mac/#documentation/Cocoa/Conceptual/ObjectiveC/Introduction/introObjectiveC.html>
- [19] SOFTENG. “Metodología Scrum”. [Documento online]. Disponible en: <http://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>
- [20] Institute of Electronic and Electrical Engineers Press. “IEEE std 830-1993, recommended practice for software requirements specifications”. Diciembre 1993.
- [21] Méndez, G. “Especificación de Requisitos según el estándar de IEEE 830 IEEE Std. 830-1998”. [Documento online], 22/10/2008. Disponible en: <http://www.fdi.ucm.es/profesor/gmendez/docs/is0809/ieee830.pdf>
- [22] Apple Developer. “iOS Human Interface Guidelines”. [Documento online], Última actualización 19/09/2012. Disponible en: <http://developer.apple.com/library/ios/#DOCUMENTATION/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html>
- [23] Apple Developer. “App Review Guidelines”. [Documento online]. Disponible en: <https://developer.apple.com/appstore/guidelines.html>
- [24] Masó, J.; Pomakis, K.; Juliá, N. “OpenGIS Web Map Tile Service Implementation Standard | OGC®” [Documento online]. Disponible en: <http://www.opengeospatial.org/standards/wmts>
- [25] Grimes, S. “MapKit Overlays – Session 1: Overlay Map”. [Documento online], 23/10/2010. Disponible en: <http://www.shawngrimes.me/2010/12/mapkit-overlays-session-1-overlay-map/>
- [26] Apple Developers. “XCode Unit Testing Guide”. [Documento online], Última actualización 09/01/2012. Disponible en: https://developer.apple.com/library/mac/#documentation/DeveloperTools/Conceptual/UnitTesting/00-About_Unit_Testing/about.html

APÉNDICES

APÉNDICE A PRESUPUESTO DEL PROYECTO

Una vez conocido el esfuerzo necesario para la realización del proyecto, se detallar el coste del mismo. En primer lugar se computan los gastos asociados al personal que se encarga de la realización de la aplicación. Su salario varía en función del puesto que desempeñen.

Como personal se ha incluido a Adrián Casado Rivas como principal desarrollador de la aplicación. También a Javier García Guzmán, ya que es el tutor del proyecto y también se deben computar sus horas de reuniones y ayuda en el desarrollo de la aplicación. Y por último se ha incluido a Juan Carlos Alonso Durán, porque, aparte de desarrollar el servicio web del SEL-UC3M, ha aportado toda la ayuda necesaria al autor para conseguir superar el proyecto.

El coste total asociado al personal asciende a 12.985,6 €.

Personal		Categoría	Coste hombre/hora	Dedicación (Horas)	Coste (Euro)
Casado Adrián	Rivas,	Ingeniero Junior	20	600	12.000
García Javier	Guzmán,	Ingeniero Senior	36	17,6	633,6
Alonso Juan Carlos	Durán,	Ingeniero Junior	20	17,6	352
Hombre hora			76	Total	12.985,6

Tabla 6.1: Coste asociado al personal

También se deben tener en cuenta como gastos asociados a la realización del proyecto las amortizaciones. En este caso las amortizaciones son aplicables al ordenador de sobremesa utilizado para el proyecto, la licencia de desarrollador y el terminal móvil utilizado.

Se ha considerado un tiempo de vida útil para los ordenadores de 4 años, mientras que para el móvil se han considerado 3 años de tiempo de vida útil. La licencia de desarrollador se renueva anualmente.

El coste imputable se ha calculado utilizando los siguientes datos:

- **A** = número de meses desde la fecha de facturación en que el equipo es utilizado
- **B** = periodo de deprecación (48 o 36 meses dependiendo del tipo de dispositivo)
- **C** = coste del equipo (sin IVA)
- **D** = % de uso que se dedica al proyecto

Teniendo en cuenta los anteriores datos, el coste imputable se calcula mediante la siguiente fórmula de cálculo de amortización:

$$A/B * C * D$$

El coste total asociado a las amortizaciones es de 554,83 €.

Producto	Uso del producto	Periodo de amortización	Precio	Dedicación a proyecto	Coste para el proyecto
iMac 21.5'	8	48	1.145 €	100%	190,83 €
Licencia de Desarrollador iOS	8	12	80 €	100%	53,33 €
iPhone 4S	8	18	699 €	100%	310,66 €
TOTAL					554,83 €

Tabla 6.2: Coste de Amortizaciones

El coste final del desarrollo de la aplicación se obtiene realizando la suma del coste de personal y el coste de las amortizaciones. La cantidad total asciende a 13.540,43 € (trece mil quinientos cuarenta euros con cuarenta y tres céntimos).

Presupuesto	Coste (Euro)
Coste personal	12.985,6
Coste amortizaciones	554,83
Coste Total	13.540,43

Tabla 6.3: Coste total

APÉNDICE B MANUAL DE USUARIO

Este manual de usuario pretende ser una guía rápida para que comencemos a utilizar la aplicación UC3M Mapas en su versión para los dispositivos iOS.

En primer lugar, es necesario bajar la aplicación introduciendo en la App Store, instalada por defecto en el dispositivo, el nombre de la app, UC3M Mapas. También es posible acceder al enlace de descarga mediante el código QR realizado al efecto. El QR lleva a un enlace que detecta el sistema operativo del dispositivo y redirige a la tienda de apps correspondiente.

Al abrir por primera vez la app verá la vista principal de la aplicación. En la parte superior se encuentra la barra de búsqueda donde podrá introducir el término que desee. Previamente deberá configurar la app para que la aplicación realice las búsquedas con sus preferencias.

Para configurar la app, primeramente en la vista principal deberá seleccionar el tipo de búsqueda a realizar, de lugar o de persona, utilizando las opciones del botón segmentado de la parte inferior. Después deberá pulsar el botón de configuración, en la esquina superior derecha de la vista principal, para seleccionar el campus sobre el que realizar las búsquedas y el tipo de mapa a visualizar.

Una vez configurada la app, introduzca en la barra de búsqueda de la vista principal el término a buscar y pulse sobre el botón de búsqueda. Aparecerá en el mapa la ubicación del lugar o la persona sobre el plano de la planta del edificio en el que se encuentra la ubicación.

Podrá pulsar sobre el globo de texto para cerrar el globo de texto que aparece sobre la ubicación o pulsar sobre el botón que tiene para acceder a más información del lugar o de la persona buscada. Si pulsa el botón del globo de texto, accederá a la vista de detalle del lugar o de la persona y en este último caso, podrá interactuar con los datos de contacto de la persona para ponerse en contacto con la misma desde la app.

Si desea visualizar los créditos y la ayuda de la aplicación, pulse el botón de información situado en la esquina superior izquierda de la vista principal. Se le mostrará una vista con toda la información acerca de la aplicación y los desarrolladores junto a información de soporte.

El equipo de miSEL desea que disfrute de la app y consiga sacarle el mayor partido. Para cualquier consulta o sugerencia puede ponerse en contacto con nosotros en la dirección de correo electrónico sugerencias_apps@uc3m.es y para cualquier incidencia en inci_despachos_apps@uc3m.es.



1. Botón de información
2. Botón de configuración
3. Barra de búsqueda
4. Barra de indicación de ubicación
5. Botón de posicionamiento
6. Barra de elección de tipo de búsqueda

Figura 6.1: Vista principal



7. Botón de volver a la vista principal
8. Botón de reporte de incidencias
9. Detalle de la búsqueda

Figura 6.2: Vista detalle de búsqueda